Multiplicative noise removal using L1 fidelity on frame coefficients

Sylvain Durand, Jalal M. Fadili, Mila Nikolova

To appear in Journal of Mathematical Imaging and Vision

Multiplicative Noise Removal Using L1 Fidelity on Frame Coefficients

Durand S.*, Fadili J.† and Nikolova M.◇

* M.A.P. 5, Université Descartes (Paris V), 45 rue des Saint Pères, 75270 Paris Cedex 06, France
email: Sylvain.Durand@mi.parisdescartes.fr
† GREYC CNRS-ENSICAEN-Universié de Caen 6, Bd Maréchal Juin 14050 Caen Cedex, France
email: Jalal.Fadili@greyc.ensicaen.fr
◇ CMLA, ENS Cachan, CNRS, PRES UniverSud, 61 Av. President Wilson, 94230 Cachan, France
email: Mila.Nikolova@cmla.ens-cachan.fr
(alphabetical order of the authors)

Keywords: Curvelets; Douglas-Rachford splitting; ℓ^1 data-fidelity; Multiplicative noise removal; Nonsmooth optimization; Proximal calculus; Tight frames; TV regularization; Variational methods.

Abstract

We address the denoising of images contaminated with multiplicative noise, e.g. speckle noise. Classical ways to solve such problems are filtering, statistical (Bayesian) methods, variational methods, and methods that convert the multiplicative noise into additive noise (using a logarithmic function), apply a variational method on the log data or shrink their coefficients in a frame (e.g. a wavelet basis), and transform back the result using an exponential function.

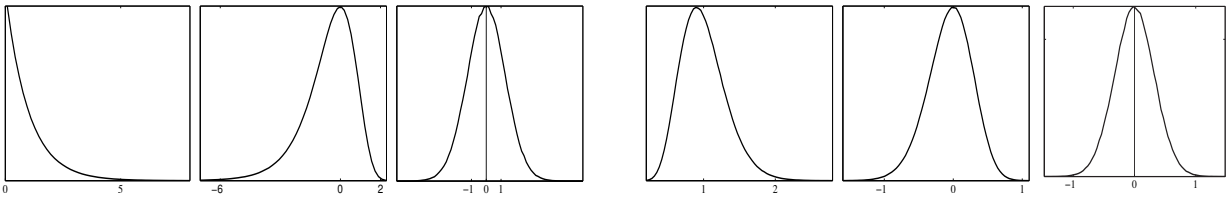
We propose a method composed of several stages: we use the log-image data and apply a reasonable under-optimal hard-thresholding on its curvelet transform; then we apply a variational method where we minimize a specialized hybrid criterion composed of an ℓ^1 data-fidelity to the thresholded coefficients and a Total Variation regularization (TV) term in the log-image domain; the restored image is an exponential of the obtained minimizer, weighted in a such way that the mean of the original image is preserved. Our restored images combine the advantages of shrinkage and variational methods and avoid their main drawbacks. Theoretical results on our hybrid criterion are presented. For the minimization stage, we propose a properly adapted fast scheme based on Douglas-Rachford splitting. The existence of a minimizer of our specialized criterion being proven, we demonstrate the convergence of the minimization scheme. The obtained numerical results clearly outperform the main alternative methods especially for images containing tricky geometrical structures.

1 Introduction

In various active imaging systems, such as synthetic aperture radar (SAR), laser or ultrasound imaging, the data representing the underlying (unknown image) $S_o : \Omega \to \mathbb{R}_+$, $\Omega \subset \mathbb{R}^2$, are corrupted with multiplicative noise. Such a noise severely degrades the image (see Fig. 2(a)-(b) as well as the noisy images in section 6). In order to increase the chance to restore a better image, several independent measurements for the same scene should be realized, thus yielding a set of data:

$$\Sigma_k = S_o\, \eta_k + n_k, \quad \forall k \in \{1, \cdots, K\}, \quad K \geq 1, \tag{1}$$

where $\eta_k : \Omega \to \mathbb{R}_+$, and n_k represent the multiplicative and the additive noise relevant to each measurement k. The additive noise n_k has usually a very weak effect and is systematically neglected in



(a) pdf($\eta = \eta_k$) (b) pdf(n) (c) Hist(Wn) (d) pdf(η) (e) pdf(n) (f) Hist(Wn)

Figure 1: Noise distributions: $\underline{K = 1}$ in (a), (b), (c) and $\underline{K = 10}$ in (d), (e), (f). In (c) and (f) W is a curvelet frame. Hist in (c) and (f) stands for histogram.

the literature. A commonly used realistic model for the distribution of η_k is the one-sided exponential distribution, shown in Fig. 1(a):

$$\eta_k \quad : \quad \mathrm{pdf}(\eta_k) = \mu\, e^{-\mu\eta_k}\, \mathbb{1}_{\mathbb{R}_+}(\eta_k) \quad \text{where} \quad \mathbb{1}_{\mathbb{R}_+}(t) \stackrel{\text{def}}{=} \begin{cases} 1 & \text{if } t \geq 0, \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

The data used for denoising is the average of the set of all K measurements that could be realized (see e.g. Fig. 2(b)). In classical SAR modeling one takes[1] $\mu = 1$ in (2), so the usual data production model reads (see e.g. [3, 67, 68, 72] among many other references):

$$\Sigma = \frac{1}{K}\sum_{k=1}^{K} \Sigma_k = S_o\, \frac{1}{K}\sum_{k=1}^{K} \eta_k = S_o\, \eta\ . \tag{3}$$

Since all η_k are independent, the resultant mean of the multiplicative noise η in (3) follows a Gamma distribution, see Fig. 1(d) for $K = 10$:

$$\eta = \frac{1}{K}\sum_{k=1}^{K} \eta_k : \quad \mathrm{pdf}(\eta) = \frac{K^K \eta^{K-1}}{\Gamma(K)} e^{-K\eta}, \tag{4}$$

where Γ is the usual Gamma-function and pdf stands for probability density function.

Conversion of the multiplicative noise into additive noise. A large variety of methods—references are given in § 1.1—rely on the conversion of the multiplicative noise into additive noise using

$$v = \log \Sigma = \log S_o + \log \eta = u_o + n. \tag{5}$$

In this case the probability density function of n reads (see Fig. 1(b)-(e)):

$$n = \log \eta : \quad \mathrm{pdf}(n) = \frac{K^K e^{K(n-e^n)}}{\Gamma(K)}\ . \tag{6}$$

In our experiments, we will consider noisy images for $K = 1$ (see e.g. Fig. 2(a)), in which case pdf(η) and pdf(n) are trivial to derive from (4) and (6).

1.1 Preexisting restoration methods

Various adaptive filters for multiplicative noise removal have been proposed, see e.g. [41,73] and references therein. Experiments have shown that filtering methods work well when the multiplicative noise is weak, i.e. when K is large. However, in practice it is seldom possible to get a large K.

[1] Taking $\mu \neq 1$ just amounts to re-scale S by μ.

1.1.1 Bayesian MAP, regulrization and variational methods

Even though the methods belonging to this class are inspired by different motivations, it is well known that given v, the general approach define the restored function $\hat{u}$ as

$$\hat{u} = \underset{u}{\operatorname{argmin}}\ \mathcal{F}_v(u),$$

where u is defined on a continuous or a finite domain $\Omega \subset \mathbb{R}^2$, $\mathcal{F}_v$ is a criterion combining data-fidelity Ψ with priors (regularization) Φ balanced via a parameter $\rho > 0$:

$$\mathcal{F}_v(u) = \rho\Psi(u) + \Phi(u) \ \text{ for } \ \left\{ \begin{array}{rcl} \Psi(u) & = & \displaystyle\int_\Omega \psi\big(u(\xi), v(\xi)\big)d\xi, \\ \Phi(u) & = & \displaystyle\int_\Omega \phi(|\nabla u(\xi)|)\, d\xi, \end{array} \right. \quad \xi = (\xi_1, \xi_2) \in \Omega. \tag{7}$$

In what follows, the overscript " $\hat{\ }$ " denotes a minimizer of a given criterion and when there is no ambiguity, the corresponding restored image. In (7), $\psi : \mathbb{R}_+ \to \mathbb{R}_+$ assesses closeness to data, ∇ stands for gradient (possibly in a distributional sense or a discrete approximation when Ω is finite), $\phi : \mathbb{R}_+ \to \mathbb{R}_+$ is an increasing function modeling the prior on $|\nabla u|$ and $|\cdot|$ is a norm on the gradient field. Rudin, Osher and Fatemi [61] proved that when Ω is finite, $\phi(t) = t$ yields solutions $\hat{u}$ that preserve edges. The resultant (prior) regularization term, known as Total Variation (TV), is very popular. It reads

$$\|u\|_{\mathrm{TV}} \stackrel{\mathrm{def}}{=} \int_\Omega |\nabla u(\xi)|\, d\xi,$$

where $|\nabla u| = \sqrt{(\partial u/\partial\xi_1)^2 + (\partial u/\partial\xi_2)^2}$.

Next we sketch the main modern MAP or variational methods that were used for multiplicative noise removal. Such methods were applied to raw-data (3)-(4) as well as to log-data (5)-(6). In [4], the authors develop a Bayesian MAP estimator on the log-image where the data-fitting term Ψ is derived from (6) and a specially designed prior term. The method of Aubert-Aujol (AA) [8] uses the raw data (3)-(4) and Ψ is the log-likelihood derived from (4). The criterion reads

$$\text{(AA)} \qquad \mathcal{F}_\Sigma(S) = \rho \int \left(\log S(\xi) + \frac{\Sigma(\xi)}{S(\xi)} \right) d\xi + \|S\|_{\mathrm{TV}}. \tag{8}$$

The method proposed in [40] combines the log-likelihood associated to (6) and a smoothed TV regularization. A simple and fast method using graph-cut minimization is proposed in [29].

The methods in [59, 60] consider white Gaussian multiplicative noise with mean 1 and a very small variance. In [60], Rudin Lions and Osher (RLO) minimize a criterion of the form

$$\mathcal{F}_\Sigma(S) = \rho \left\| \frac{\Sigma}{S} - 1 \right\|_2^2 + \|S\|_{\mathrm{TV}}\ . \tag{9}$$

The methods proposed by Shi and Osher (SO) in [62] deal with various noises and provide iterative TV regularization using relaxed inverse scale space (RISS) flows. The first one uses the log-data v and is based on the criterion $\mathcal{F}_v(u) = \rho\|u - v\|_2^2 + \|u\|_{\mathrm{TV}}$. The corresponding RISS flow reads

$$\text{(SO)} \qquad \begin{array}{rcl} u_t & = & \operatorname{div}\left(\dfrac{\nabla u}{|\nabla u|} \right) + \rho(v - u + v), \\ v_t & = & \alpha(v - u), \quad \alpha > 0, \quad \text{with} \quad v_0 = 0 \ \text{ and } \ u_0 = \operatorname{mean}(v). \end{array} \tag{10}$$

The second one generalizes several multiplicative noise models in the image domain [8, 28, 60] and the RISS flow uses iterative TV regularization on $\exp(S)$ for convergence reasons. According to the authors, both methods provide similar solution qualities. We focus on the first one mainly because of its simplicity.

Whatever smooth data-fidelity is chosen, it was proven [49, 52] that TV regularization yields images containing numerous constant regions (the well known stair-casing effect), hence textures and fine details are removed; this is clearly visible in the restorations obtained using these methods as reported in section 6. The results of [61] initiated a flood of papers to construct smooth edge-preserving convex functions ϕ, see e.g. [2, 21, 69], and [9] for a recent overview. Even though smoothness at zero alleviates stair-casing, it was proven in [53] that these functions ϕ lead to images with underestimated edges. This is particularly annoying if the sought-after image has neat edges or spiky features since the latter are eroded. Nevertheless, the good point is that these methods enable the introduction of priors in the image domain.

1.1.2 Multiscale shrinkage for the log-data

Many authors—e.g. [3, 5, 37, 55, 72] and references therein—focus on restoring the log-data v in (5) using decompositions into some multiscale frame for $L^2(\mathbb{R}^2)$. Let us remind that the analysis operator W of a frame $\mathcal{W} = \{w_i : i \in I\}$, where I is a set of indices, i.e. $(Wz)[i] = \langle z, w_i \rangle$, $\forall i \in I$, satisfies the generalized Parseval condition $c_1 \|z\|_2^2 \leq \sum_{i\in I} |\langle z, w_i\rangle|^2 \leq c_2 \|z\|_2^2$, $\forall z \in L^2(\mathbb{R}^2)$, where $c_2 \geq c_1 > 0$ are the frame bounds. When $c_1 = c_2$, the frame is said to be tight. In particular $\mathcal{W}$ can be a wavelet basis. The frame is chosen so that u_o admits a sparse representation, i.e. $u_o \approx \sum_{i\in J} \zeta_i w_i$ for some $\zeta_i \in \mathbb{R}$ with $\#J \ll \#I$.

The usual strategy is to decompose the log-data v into an well-adapted W:

$$y = Wv = Wu_o + Wn \ . \tag{11}$$

The rationale is that the noise Wn in y approaches a Gaussian distribution—see Figs. 1(c) and (f)—according to the Central Limit Theorem. The tendency to normality gets better as K increases. Under different frameworks, coefficients are denoised using shrinkage estimators:

$$y_{\mathcal{T}}[i] = \mathcal{T}\big((Wv)[i]\big), \quad \forall i \in I, \tag{12}$$

where $\mathcal{T} : \mathbb{R} \to \mathbb{R}$ is a symmetric function satisfying $0 \leq \mathcal{T}(t) \leq t$, $\forall t \geq 0$. The most widely used examples are soft and hard thresholding inaugurated in [30], and are given in (16) and (36) later on. Since then, various shrinkage functions $\mathcal{T}$ have been explored, e.g. [6, 11, 31, 47, 64, 70] to name only a few. See also those derived in a Bayesian framework and applied to multiplicative noise removal e.g. in [3, 5, 72].

Let $\widetilde{\mathcal{W}} = \{\widetilde{w}_i : i \in I\}$ be a dual frame. The associated dual operator $\widetilde{W} : y \mapsto \sum_{i\in I} y[i]\, \widetilde{w}_i$ is a left inverse of W, that is $v = \sum_{i\in I} (Wv)[i]\, \widetilde{w}_i$. Then a denoised log-image $v_{\mathcal{T}}$ is generated by expanding the shrunk coefficients $y_{\mathcal{T}}$ in the dual frame reads

$$v_{\mathcal{T}} = \sum_{i\in I} \mathcal{T}((Wv)[i])\, \widetilde{w}_i = \sum_{i\in I} \mathcal{T}(y[i])\, \widetilde{w}_i, \tag{13}$$

and the sought-after image reads $S_{\mathcal{T}} = \exp(v_{\mathcal{T}})$.

The major problems with shrinkage denoising methods, as sketched in (12)-(13), is that shrinking large coefficients entails an erosion of the spiky image features, while shrinking small coefficients towards

zero yields Gibbs-like oscillations in the vicinity of edges and a loss of texture information. On the other hand, if shrinkage is not strong enough, some coefficients bearing mainly noise will remain almost unchanged—we call such coefficients *outliers*—and (13) suggests they generate artifacts with the shape of the functions $\widetilde{w}_i$. An illustration can be seen in Fig. 2(b-h). Several improvements, such as translation invariant thresholding [23] and block-Stein thresholding (BS) [22], were brought in order to alleviate these artifacts. Results obtained using the BS method are presented in section 6: the above mentioned artifacts remain visible. Another inherent difficulty comes from the fact that coefficients between different scales are not independent, as usually assumed, see e.g. [7,11,47,63]. In summary, as shrinkage-based estimation relies on sparsity of the representation, it is able to capture efficiently faint structures in the image. But this comes at the price of an intricate choice of the shrinkage function and the associated parameters (e.g. threshold).

1.2 Our approach is hybrid

We initially restore the log-data (5) and then derive the restored image $\hat{S}$. Our objective is to avoid the main drawbacks of variational and sparsity-based shrinkage methods and to take benefit of the best of both worlds. A way to achieve such a goal is to combine the information brought by the coefficients of the frame-transformed data along with pertinent regularization in the domain of the log-image. This idea for the purpose of additive Gaussian noise removal have been investigated in several papers [13,15,20,24,33, 36,44,45].

Although guided by different arguments, hybrid methods amount to define the restored function $\hat{u}$ as

$$\hat{u} \in \underset{u}{\operatorname{argmin}} \ \Phi(u) \quad \text{subject to} \quad |(W(u-v))[i]| \leq \mu_i, \, \forall i \in I \ .$$

If the use of an edge-preserving regularization, such as TV for Φ is a pertinent choice, the strategy for the selection of parameters $(\mu_i)_{i\in I}$ is more tricky since it must take into account the magnitude of the data coefficients $(y[i])_{i\in I}$. However, deciding on the value of μ_i based solely on $y[i]$, as done in these papers, is too rigid since there are either correct data coefficients that incur smoothing ($\mu_i > 0$), or noisy coefficients that are left unchanged ($\mu_i = 0$). A way to deal with this situation is to determine $(\mu_i)_{i\in I}$ based both on the data and on the prior term Φ. According to the theoretical results derived in [50,51], this objective can be carried out by defining a non-smooth data-fitting term for the coefficients, as done by some of the authors of this paper in [32] which gave rise to very successful numerical results.

To the best of our knowledge, hybrid methods have never been applied to multiplicative noise removal before, whereas the latter is a challenging problem: it arises in important applications but up to now, there is no entirely satisfactory methods to solve it.

We propose a method where the restored log-image $\hat{u}$ is defined as the minimizer of a criterion composed of an ℓ^1-fitting to the (suboptimally) hard-thresholded frame coefficients of the log-data and a TV regularization in the log-image domain (section 2). The rationale behind this choice and several theoretical properties are presented as well. This method uses some ideas from a previous work of some of the authors [32]. The minimization scheme to compute the log-restored image uses a Douglas-Rachford splitting scheme specially adapted to our criterion (section 3). It involves original derivations and proofs. The sought-after image is of the form $B\exp(\hat{u})$ where B is a bias-correction term derived from the noise

distribution (section 4). The full algorithm to remove multiplicative noise is summarized in section 5. A large variety of experiments and comparisons to other contemporary multiplicative noise removal methods are presented in section 6. Concluding remarks and open questions are discussed in section 7.

2 Restoration of the frame coefficients of the log-data

In this section we consider how to restore the frame coefficients $y = Wv$ of the log-data image v obtained according to (5). We focus on methods which, for a given preprocessed data set, lead to convex optimization problems. The denoised coefficients are denoted by $\hat{x}$.

We assume that $v \in L^2(\Omega)$ which ensures that $y = Wv \in \ell^2(I)$.

2.1 Specific requirements to restore the coefficients

Given the log-data v obtained according to (5), we first apply a frame transform as in (11) to get

$$y[i] = \langle w_i, v\rangle = \langle w_i, u_o\rangle + \langle w_i, n\rangle, \quad \forall i \in I, \tag{14}$$

where u_o denotes the unknown original log-image. The noise contained in the i-th datum reads $\langle n, w_i\rangle$; its distribution is of the form displayed in Fig. 1(c) or (f). However the signal to noise ratio (SNR) of the coefficients is ill-assorted. When u_o has a sparse representation in the frame, many coefficients contain only noise. For this reason, we apply a hard-thresholding to all coefficients

$$y_{\mathcal{T}_{\rm H}}[i] \stackrel{\rm def}{=} \mathcal{T}_{\rm H}\big(y[i]\big), \quad \forall i \in I, \tag{15}$$

where the hard-thresholding operator $\mathcal{T}_{\rm H}$, with a threshold T, reads [30]

$$\mathcal{T}_{\rm H}(t) = \begin{cases} 0 & \text{if } |t| \le T, \\ t & \text{otherwise.} \end{cases} \tag{16}$$

The resultant set of coefficients is systematically denoted by $y_{\mathcal{T}_{\rm H}}$. We choose an *underoptimal* threshold T in order to preserve as much as possible the information relevant to edges and to textures, an important part of which is contained in small coefficients. Let's point out that with hard thresholding, all kept coefficients are unaltered and thus contain the original information on the sought-after image.

Let us consider

$$v_{\mathcal{T}_{\rm H}} = \sum_{i\in I} y_{\mathcal{T}_{\rm H}}[i]\, \widetilde{w}_i = \sum_{i\in I_1} y[i]\, \widetilde{w}_i, \tag{17}$$

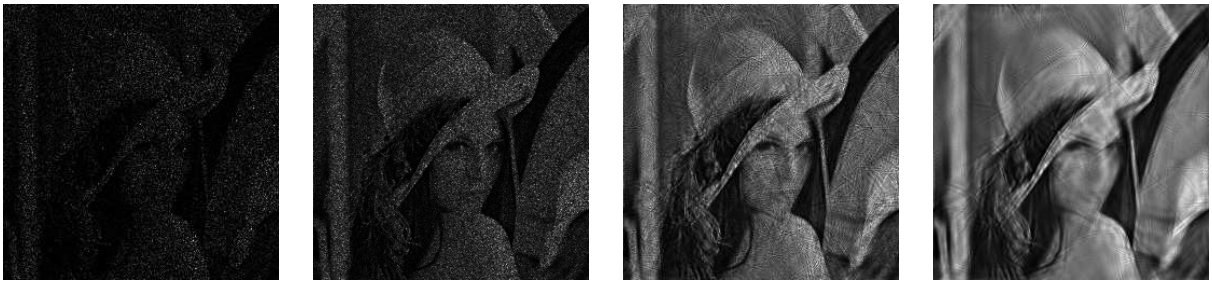
where

$$I_1 = \{i \in I \,:\, |y[i]| > T\}. \tag{18}$$

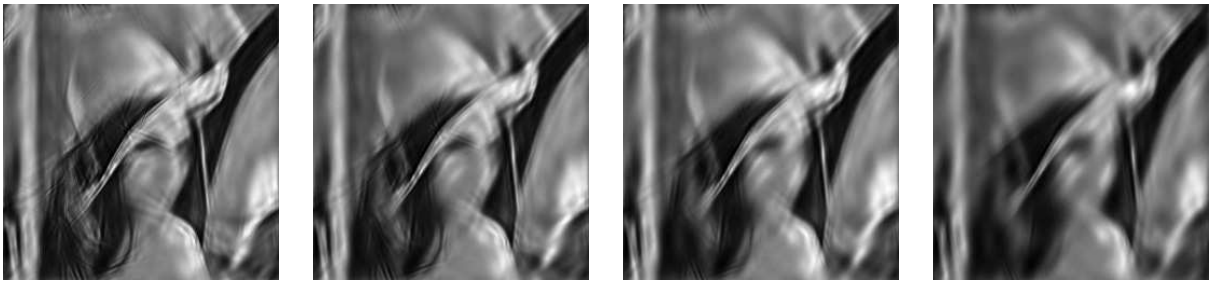
The image $v_{\mathcal{T}_{\rm H}}$ contains artifacts with the shape of the $\widetilde{w}_i$'s, for all $y[i]$, that are dominated by noise and above the threshold T, as well as a lot of information about the fine details in the original (unknown) log-image u_o. In all cases, whatever the choice of T, the image $v_{\mathcal{T}_{\rm H}}$ is unsatisfactory—see Fig. 2 (c)-(h).

Given the under-thresholded data $y_{\mathcal{T}_{\rm H}}$, we focus on hybrid methods where the restored coefficients $\hat{x}$ minimize a function $F_y : \ell^2(I) \to \mathbb{R} \cup \{\infty\}$ of the form:

$$\begin{cases} \hat{x} & = & \displaystyle\operatorname*{argmin}_{x\in\ell^2(I)} F_y(x) \\ \\ F_y(x) & = & \Psi(x, y_{\mathcal{T}_{\rm H}}) + \Phi(\widetilde{W}x), \end{cases} \tag{19}$$



(a) Noisy for $K = 1$ (b) Noisy for $K = 4$ (c) $T = 2\sigma$ (d) $T = 3\sigma$



(e) $T = 4\sigma$ (f) $T = 5\sigma$ (g) $T = 6\sigma$ (h) $T = 8\sigma$

Figure 2: (a) Noisy Lena for $K = 1$. (b) Noisy Lena obtained via averaging, see (1), for $K = 4$. (c)-(h) Denoising of data v shown in (b) where the curvelet transform of v are hard-thresholded according to (15)-(17) for different choices of T where (see (61)). The displayed restorations correspond to $v_{\mathcal{T}_{\mathrm{H}}} = \exp(v_{\mathcal{T}_{\mathrm{H}}})$, as given in (17).

where Ψ is a data-fidelity term in the domain of the frame coefficients and Φ is an edge-preserving regularization term in the log-image domain. The restored log-image $\hat{u}$ is of the form

$$\hat{u} = \widetilde{W}\hat{x} \ . \tag{20}$$

Let us denote

$$I_0 = I \setminus I_1 = \{i \in I \, : \, |y[i]| \leq T\}, \tag{21}$$

where I_1 is given in (18). In order to specify the shape of Ψ and Φ, we analyze the information borne by the coefficients $y_{\mathcal{T}_{\mathrm{H}}}[i]$ relevant to I_0 and to I_1.

(I_0) The coefficients $y[i]$ for $i \in I_0$ usually correspond to high-frequency components which can be of the two types described below.

(a) Coefficients $y[i]$ containing essentially noise—in which case the best we can do is to keep them null, i.e. $\hat{x}[i] = y_{\mathcal{T}_{\mathrm{H}}}[i] = 0$;

(b) Coefficients $y[i]$ which correspond to faint edges and details in u_o. Since $y[i]$ is difficult to distinguish from the noise, the relevant $\hat{x}[i]$ should be restored using the edge-preserving prior conveyed by Φ. Let us emphasize that a careful restoration must find a nonzero $\hat{x}[i]$, since otherwise $\hat{x}[i] = 0$ would generate Gibbs-like oscillations in $\hat{u}$.

(I_1) The coefficients $y[i]$ for $i \in I_1$ are of the following two types:

(a) Large coefficients which carry the main features of the sought-after function $\hat{u}$. They verify $y[i] \approx \langle w_i, u_o \rangle$ and can be kept intact, i.e. $\hat{x}[i] = y_{\mathcal{T}_\mathrm{H}}[i] = y[i]$.

(b) Coefficients which are highly contaminated by noise, characterized by $|y[i]| \gg |\langle w_i, u_o \rangle|$. We call them outliers: if we had $\hat{x}[i] = y[i]$, by (17) we get $v_{\mathcal{T}_\mathrm{H}} = \sum_{j \neq i} \hat{x}[j] \widetilde{w}_j + y[i] \widetilde{w}_i$ which shows that $\hat{u}$ would contain an artifact with the shape of $\widetilde{w}_i$. For this reason, $\hat{x}[i]$ must be restored according to the prior Φ.

2.2 A specialized hybrid criterion

This analysis clearly defines the goals that the minimizer $\hat{x}$ of F_y is expected to achieve. In particular, $\hat{x}$ must involve an implicit classification between coefficients that fit to $y_{\mathcal{T}_\mathrm{H}}$ exactly and coefficients that are restored according to the prior term Φ. In short, restored coefficients have to fit $y_{\mathcal{T}_\mathrm{H}}$ exactly if they are in accordance with the regularization term Φ and have to be restored via the latter otherwise. Since [50, 51] it is known that criteria F_y where Ψ is non-smooth at the origin (e.g. ℓ^1) can satisfy $\hat{x}[i] = y_{\mathcal{T}_\mathrm{H}}[i]$ for coefficients that are in accordance with the prior Φ, while the other coefficients are restored according to Φ, see also [32]. For these reasons, we focus on a criterion of the form (19) where

$$\Psi(x) = \sum_{i \in I_1} \lambda_i \, |(x-y)[i]| + \sum_{i \in I_0} \lambda_i \, |x[i]| = \|\Lambda(x - y_{\mathcal{T}_\mathrm{H}})\|_1 \ , \tag{22}$$

$$\text{for} \qquad \Lambda \stackrel{\text{def}}{=} \mathrm{diag}\{\lambda_i : i \in I\} \ , \tag{23}$$

$$\Phi(x) = \int_\Omega \left| \nabla \widetilde{W} x \right| \, d\xi = \left\| \widetilde{W} x \right\|_{\mathrm{TV}} \ . \tag{24}$$

Remark 1 *We should emphasize that the TV regularization term Φ in our criterion will not favor minimizers $\hat{u} = \widetilde{W}\hat{x}$ that involve constant regions unlike the usual variational methods, as discussed in § 1.1.1. The reason is that the non differentiability of Ψ at $y_{\mathcal{T}_\mathrm{H}}$ ensures that $\hat{x}[i] = y_{\mathcal{T}_\mathrm{H}}[i]$ for a certain number of coefficients [50]. Then $\widetilde{W}\hat{x}$ keeps some fixed structures of the form $y_{\mathcal{T}_\mathrm{H}}[i]\widetilde{w}_i$ which can prevent from stair-casing in $\hat{u}$, provided that there is no large sets of outliers corresponding to the same region of the image all of which are restored according to the TV term Φ. Such an important benefit clearly depends on the level of the noise, the threshold T and the choice of $(\lambda_i)_{i \in I}$.*

2.3 Well-posedness of the minimization problem

The theorem below ensures the existence of a minimizer of the criterion F_y defined above. Its proof can be found in [32].

Theorem 1 [32] *For $y \in \ell^2(I)$ and $T > 0$ given, consider F_y as defined by (15), (19) and (22)-(24), where $\Omega \in \mathbb{R}^2$ is open, bounded and its boundary $\partial\Omega$ is Lipschitz, and $\mathcal{W} = \{w_i : i \in I\}$ is a frame of $L^2(\Omega)$. Suppose that*

1. *$\widetilde{W}$ is the pseudo inverse of W, i.e. $\widetilde{W} = (W^*W)^{-1}W^*$ where W^* is the adjoint operator;*
2. *$\lambda_{\min} = \min_{i \in I} \lambda_i > 0$.*

Then F_y has a minimizer $\hat{x} \in \ell^2(I)$.

Remark 2 *Since $\mathcal{W}$ is a multiscale frame for $L^2(\mathbb{R}^2)$ and $\hat{x} \in \ell^2(I)$, the restored log-image $\hat{u}$ as given in* (20) *satisfies $\hat{u} \in L^2(\Omega)$. Note that assumption 1 holds true in many cases, e.g. for tight frames this choice of $\widetilde{W}$ is standard since it leads to $\widetilde{W} = c^{-1}W^*$, where $c \in (0, \infty)$ is a constant.*

Let us notice that the minimizer of F_y is not necessarily unique. Given y, we denote by $\mathcal{G}_y$ the set of all minimizers of F_y:

$$\mathcal{G}_y \stackrel{\text{def}}{=} \big\{\hat{x} \in \ell^2(I) \ : \ F_y(\hat{x}) = \min_x F_y(x)\big\}. \tag{25}$$

For every $y_{\mathcal{T}_\mathrm{H}}$, the set $\mathcal{G}_y$ is convex and corresponds to images $\hat{u} = \widetilde{W}\hat{x}$ which are visually very similar, as stated in the theorem below whose proof can be found in [32].

Theorem 2 [32] *Let the assumptions of Theorem 1 hold. If $\hat{x}_1$ and $\hat{x}_2$ are two minimizers of F_y (i.e. $\hat{x}_1 \in \mathcal{G}_y$ and $\hat{x}_2 \in \mathcal{G}_y$), then*

$$\nabla\widetilde{W}\hat{x}_1 \propto \nabla\widetilde{W}\hat{x}_2, \qquad \text{a.e. on } \Omega,$$

i.e. $\widetilde{W}\hat{x}_1$ and $\widetilde{W}\hat{x}_2$ have the same level lines.

In words, the images $\hat{u}_1 = \widetilde{W}\hat{x}_1$ and $\hat{u}_2 = \widetilde{W}\hat{x}_2$ share the same level lines, i.e. they are obtained from each other by a local change of contrast. This is usually invisible to the naked eye.

Some guidelines for the choice of λ_i were investigated in [32]. The conclusions can be summarized as follows.

(a) If $i \in I_1$, the parameter λ_i should be close to, but below the upper bound $\|\widetilde{w}_i\|_{\mathrm{TV}}$, since above this bound, outliers $y[i]$ cannot be restored.

(b) For $i \in I_0$, a reasonable choice is

$$\lambda_i = \max_{k \neq i} \left| \int_\Omega (\nabla \widetilde{w}_i(\xi))^T \, \frac{\nabla \widetilde{w}_k(\xi)}{|\nabla \widetilde{w}_k(\xi)|} \, d\xi \right| ,$$

where $.^T$ stands for transposed. If λ_i is below this bound, some neighboring coefficients that are erroneously set to zero might not be restored correctly, even though Gibbs-like oscillations are reduced.

Another important remark is that, for some multiscale transforms, the bounds discussed above are constant. This is for instance the case for the wavelet transform.

For the frame $\mathcal{W} = \big\{w_i : i \in I\big\}$ we here focus on *the second generation curvelet transform* because of the following facts.

- This transform is known to provide near-optimal non-adaptive sparse representation of piecewise smooth images away from smooth edges. Thus it is a very good candidate to capture efficiently the geometrical content of the log-data.
- The bounds on $(\lambda_i)_{i \in I}$ are nearly constant, so we use only two values for λ_i,

  $$\lambda_i = \tilde{\lambda}_0 > 0, \quad \forall i \in I_0 \ \text{ and } \ \lambda_i = \tilde{\lambda}_1 > 0, \quad \forall i \in I_1.$$

  Then the diagonal matrix Λ in (23) satisfies $\Lambda[i,i] \in \{\tilde{\lambda}_0, \tilde{\lambda}_1\}$.
- This curvelet transform corresponds to a tight frame which will turn out to be helpful for the subsequent optimization scheme, see in particular § 3.2.2.

3 Minimization scheme

Let us rewrite the minimization problem defined by (15), (19) and (22)-(24) in a more compact form:

$$\text{find} \quad \hat{x} \quad \text{such that} \quad F_y(\hat{x}) = \min_x F_y(x) \quad \text{for} \quad F_y(x) = \Psi(x) + \Phi(x) \ . \tag{26}$$

Clearly, Ψ in (22) and Φ in (24) are proper lower-semicontinuous convex functions, hence the same holds true for F_y. The set $\mathcal{G}_y$ introduced in (25) is non-empty by Theorem 1 and can be rewritten as

$$\mathcal{G}_y = \{\hat{x} \in \ell^2(I) \big| \hat{x} \in (\partial F_y)^{-1}(0)\},$$

where ∂F_y stands for subdifferential operator. Minimizing F_y amounts to solving the inclusion

$$0 \in \partial F_y(x) \ ,$$

or equivalently, to finding a solution to the fixed point equation

$$x = (\mathrm{Id} + \gamma \partial F_y)^{-1}(x) \ , \quad \forall \gamma > 0, \tag{27}$$

where $(\mathrm{Id} + \gamma \partial F_y)^{-1}$ is the *resolvent operator* associated to ∂F_y and Id is the identity map on the Hilbert space $\ell^2(I)$. The schematic algorithm resulting from (27), namely

$$x^{(k+1)} = (\mathrm{Id} + \gamma \partial F_y)^{-1}(x^{(k)}), \tag{28}$$

is a fundamental tool for finding the root of any maximal monotone operator [34, 58], such as e.g. the subdifferential of a convex function. By (28), $\gamma > 0$ can be seen as the stepsize of the algorithm. Unfortunately, the resolvent operator $(\mathrm{Id} + \gamma \partial F_y)^{-1}$ for F_y in (26) cannot be calculated in closed-form; we focus on iterative splitting methods.

3.1 Generalities on splitting methods

Splitting methods do not attempt to evaluate (28) directly; instead, they are based on separate evaluations of the resolvent operators $(\mathrm{Id} + \gamma \partial \Psi)^{-1}$ and $(\mathrm{Id} + \gamma \partial \Phi)^{-1}$. The latter are usually easier and this turns out to be true for our functionals Ψ and Φ in (26).

Splitting methods for monotone operators have numerous applications for convex optimization and monotone variational inequalities. Even though the literature is abundant, these can basically be systematized into three main classes: the forward-backward [38, 65, 66], the Douglas/Peaceman-Rachford [43], and the little-used double-backward [42, 54]. A recent theoretical overview of all these methods can be found in [25, 35]. Forward-backward can be seen as a generalization of the classical gradient projection method for constrained convex optimization, hence it inherits all its restrictions. Typically, one must assume that either Ψ or Φ is differentiable with Lipschitz continuous gradient, and the stepsizes γ must fall in a range dictated by the gradient modulus of continuity; see [27] for an excellent account. Since both Ψ and Φ are non differentiable, forward-backward splitting is not adapted to our criterion (26).

We will focus on a Douglas/Peaceman-Rachford approach since differentiability of neither of the functions Ψ or Φ is required. The derivation of our algorithm relies on the calculation of the proximity operators as defined next.

3.1.1 Proximity operators

Proximity operators were inaugurated in [46] as a generalization of convex projection operators.

Definition 1 (Moreau [46]) *Let $\varphi : \mathcal{H} \to \mathbb{R}$ be a proper, lower-semicontinuous and convex function where $\mathcal{H}$ is a Hilbert space. Then, for every $z \in \mathcal{H}$, the function $h \mapsto \varphi(h) + \|z-h\|_2^2/2$, for $h \in \mathcal{H}$, achieves its infimum at a unique point denoted by* $\mathrm{prox}_\varphi z$*, i.e.*

$$\mathrm{prox}_\varphi z = \underset{h\in\mathcal{H}}{\mathrm{argmin}} \left\{ \varphi(h) + \frac{1}{2}\|z-h\|_2^2 \right\}. \tag{29}$$

The operator $\mathrm{prox}_\varphi : \mathcal{H} \to \mathcal{H}$ *thus defined is called the proximity operator of φ.*

It will be convenient in the sequel to introduce the reflection operator

$$\mathrm{rprox}_\varphi \stackrel{\mathrm{def}}{=} 2\mathrm{prox}_\varphi - \mathrm{Id}. \tag{30}$$

By the minimality condition for (29), it is straightforward that $\forall z, p \in \mathcal{H}$ we have

$$p = \mathrm{prox}_\varphi z \iff z - p \in \partial\varphi(p). \tag{31}$$

Then, for any φ, we have $(\mathrm{Id} + \partial\varphi)^{-1} = \mathrm{prox}_\varphi$, and we can write for all $\gamma > 0$,

$$(\mathrm{Id} + \gamma\partial\Psi)^{-1} = \mathrm{prox}_{\gamma\Psi} \quad \text{and} \quad (\mathrm{Id} + \gamma\partial\Phi)^{-1} = \mathrm{prox}_{\gamma\Phi}. \tag{32}$$

One can note that (28) can also be written as $x^{(k+1)} = \mathrm{prox}_{\gamma F_y}(x^{(k)})$ for $\gamma > 0$.

3.1.2 Douglas-Rachford splitting algorithms

The Douglas/Peaceman-Rachford family is the most general preexisting class of maximal monotone operator splitting methods. Given a fixed scalar $\gamma > 0$ and a sequence $\mu_k \in (0,2)$, this class of methods can be expressed via the following recursion written in the compact form

$$\begin{aligned} x^{(k+1)} &= \left[\left(1 - \frac{\mu_k}{2}\right)\mathrm{Id} + \frac{\mu_k}{2}(2\mathrm{prox}_{\gamma\Psi} - \mathrm{Id}) \circ (2\mathrm{prox}_{\gamma\Phi} - \mathrm{Id}) \right] x^{(k)} \ , \\ &= \left[\left(1 - \frac{\mu_k}{2}\right)\mathrm{Id} + \frac{\mu_k}{2}\mathrm{rprox}_{\gamma\Psi} \circ \mathrm{rprox}_{\gamma\Phi} \right] x^{(k)} \ . \end{aligned} \tag{33}$$

Since our problem (26) admits solutions, the following result ensures that iteration (33) converges for our criterion F_y.

Theorem 3 *Let $\gamma > 0$ and $\mu_k \in (0,2)$ be such that $\sum_{k\in\mathbb{N}} \mu_k(2-\mu_k) = +\infty$. Take $x^{(0)} \in \ell^2(I)$ and consider the sequence of iterates defined by (33). Then, $(x^{(k)})_{k\in\mathbb{N}}$ converges weakly to some point $\bar{x} \in \ell^2(I)$ and* $\mathrm{prox}_{\gamma\Phi}\bar{x} \in \mathcal{G}_y$.

This theorem is a straightforward consequence of [25, Corollary 5.2]. For instance, the sequence $\mu_k = 1, \forall k \in \mathbb{N}$, satisfies the requirement of the theorem.

According to this theorem, the restored log-image $\hat{x}$, as defined by (19) and (22)-(24), reads

$$\hat{x} = \mathrm{prox}_{\gamma\Phi}\bar{x}, \tag{34}$$

where $\bar{x}$ is the output of the Douglas-Rachford algorithm (33).

3.2 Douglas-Rachford splitting algorithm for our criterion

To implement the Douglas-Rachford recursion (33) that solves our minimization problem (26), we need to compute the proximity operators $\mathrm{prox}_{\gamma\Psi}$ and $\mathrm{prox}_{\gamma\Phi}$. This is detailed in this subsection.

3.2.1 Proximity operator of Ψ

The proximity operator of $\gamma\Psi$ is established in the lemma stated below.

Lemma 1 *Let $x \in \ell^2(I)$. Then*

$$\mathrm{prox}_{\gamma\Psi}(x)[i] = y_{\mathcal{T}_{\mathrm{H}}}[i] + {\mathcal{T}_{\mathrm{S}}}^{\gamma\lambda_i}\left(x[i] - y_{\mathcal{T}_{\mathrm{H}}}[i]\right), \quad \forall i \in I \ , \tag{35}$$

where ${\mathcal{T}_{\mathrm{S}}}^{\theta}$ is the soft-thresholding operator with threshold θ

$${\mathcal{T}_{\mathrm{S}}}^{\theta}(t) = \begin{cases} 0 & \text{if } |t| \le \theta, \\ t - \theta\mathrm{sign}(t) & \text{otherwise,} \end{cases} \quad t \in \mathbb{R}. \tag{36}$$

Proof. Ψ as given in (22) is an additive separable function in each coordinate $i \in I$. Thus, solving the proximal minimization problem of Definition 1 is also separable and amounts to solving independently $\#I$ scalar minimization problems.
For any convex function φ and $t \in \mathbb{R}$, $s \in \mathbb{R}$, let $\psi(s) = \varphi(s - t)$. Thus, using (31), it is straightfroward to show that

$$p = \mathrm{prox}_{\psi}(s) \quad \Longleftrightarrow \quad p = t + \mathrm{prox}_{\varphi}(s - t) \ . \tag{37}$$

Furthermore, it is easy to find that for $\varphi(h) = \theta\,|h|$

$$\mathrm{prox}_{\varphi}(s) = {\mathcal{T}_{\mathrm{S}}}^{\theta}(s). \tag{38}$$

Applying (37) to $\psi(s) = \theta\,|s - t|$ with $\theta = \gamma\lambda_i$, $s = x[i]$ and $t = y_{\mathcal{T}_{\mathrm{H}}}[i]$ yields (35). □

Note that now

$$\mathrm{rprox}_{\gamma\Psi}(x) = 2\left(y_{\mathcal{T}_{\mathrm{H}}}[i] + {\mathcal{T}_{\mathrm{S}}}^{\gamma\lambda_i}\left(x[i] - y_{\mathcal{T}_{\mathrm{H}}}[i]\right)\right)_{i\in I} - x \ . \tag{39}$$

3.2.2 Proximity operator of Φ

Clearly, $\Phi(x) = \|\cdot\|_{\mathrm{TV}} \circ \widetilde{W}(x)$ is a pre-composition of the TV-norm with the linear operator $\widetilde{W}$. However, computing the proximity operator of Φ for an arbitrary $\widetilde{W}$ may be intractable. We then systematically assume that our frame is tight with constant $c \in (0, \infty)$; i.e. $W^*W = c\,\mathrm{Id}$. An immediate consequence is that $\widetilde{W} = c^{-1}W^*$.

The precise definition of the TV norm that we actually use depends on $\widetilde{W}x$ which belongs to $L^2(\Omega)$. Let $\mathcal{X} = L^2(\Omega) \times L^2(\Omega) \subset L^2(\mathbb{R}^2) \times L^2(\mathbb{R}^2)$ and $\langle\cdot,\cdot\rangle_{\mathcal{X}}$ be the inner product in $\mathcal{X}$. For any $q \in [1, \infty]$, we denote by $|||\cdot|||_q$ the L^q norm on $\mathcal{X}$. For any $\tau > 0$, we define $B_\infty(\tau)$ as the *closed* L^∞-ball of radius τ in $\mathcal{X}$,

$$B_\infty(\tau) \stackrel{\mathrm{def}}{=} \Big\{z = (z_1, z_2) \in \mathcal{X} : |z(\xi)| \le \tau, \forall \xi \in \Omega\Big\}. \tag{40}$$

Let us also define the set

$$C(\tau) \stackrel{\text{def}}{=} \left\{ \operatorname{div}(z) \in L^2(\Omega) : z \in \mathcal{C}_c^\infty(\Omega \times \Omega) \cap B_\infty(\tau) \right\} . \tag{41}$$

More precisely the TV regularization term we use reads

$$\|u\|_{\text{TV}} = \sup \left\{ \int_\Omega u(\xi)\, w(\xi)\, d\xi \ : \ w \in C(1) \right\} . \tag{42}$$

Let us remind that $\Phi(x) = \|\widetilde{W}x\|_{\text{TV}}$. The expression of $\text{prox}_{\gamma\Phi}$ is given in statement (i) of the next lemma while the computation scheme to solve statement (ii) is established in Lemma 3.

Lemma 2 *Let $x \in \ell^2(I)$ and $B_\infty(.)$ be as defined in (40).*

(i) We have

$$\text{prox}_{\gamma\Phi}(x) = \left(\text{Id} - W \circ \left(\text{Id} - \text{prox}_{c^{-1}\gamma\|\cdot\|_{\text{TV}}} \right) \circ \widetilde{W} \right)(x) \ ; \tag{43}$$

(ii) Furthermore,

$$\text{prox}_{c^{-1}\gamma\|\cdot\|_{\text{TV}}}(u) = u - P_{C(c^{-1}\gamma)}(u) \ , \quad \forall u \in L^2(\Omega) \ , \tag{44}$$

where $C(.)$ is defined in (41) and $P_{C(.)}$ is the orthogonal projector on $C(.)$.

Proof. Since $\widetilde{W}$ is bounded and linear, $\|\cdot\|_{\text{TV}}$ is continuous and convex, and there is $u \in L^2(\Omega)$ such that $\|u\|_{\text{TV}} < \infty$, it is clear that all assumptions required in [26, Proposition 11] are satisfied. Applying the same proposition yields statement (i).

We focus next on (ii). Note that for any $\tau > 0$, $C(\tau)$ in (41) is a closed convex subset since $B_\infty(\tau)$ is closed and convex, and the operator div is linear; thus the projection $P_{C(\tau)}$ is well defined.

Let us remind that the Legendre-Fenchel (known also as the convex-conjugate) transform of a function $\varphi : \mathcal{H} \to \mathbb{R}$, $\varphi \not\equiv \infty$, where $\mathcal{H}$ is an Hilbert space, is defined by

$$\varphi^\star(h) = \sup_{z \in \text{dom}(\varphi)} \left\{ \langle h, z \rangle - \varphi(z) \right\},$$

and that $\varphi^\star$ is a closed convex function. If φ is convex, proper and lower semi-continuous, the original Moreau decomposition [46, Proposition 4.a] tells us that

$$\text{prox}_\varphi + \text{prox}_{\varphi^\star} = \text{Id} \ . \tag{45}$$

One can see also [27, Lemma 2.10] for an alternate proof of (45).

Let $\imath_{\mathcal{S}}$ denote the indicator function of a nonempty set $\mathcal{S}$, i. e.

$$\imath_{\mathcal{S}}(z) = \begin{cases} 0 & \text{if } z \in \mathcal{S} \ , \\ +\infty & \text{otherwise} \ . \end{cases}$$

It is easy to see that from (41), (42) and the definition of the conjugate that

$$\left(c^{-1}\gamma\|\cdot\|_{\text{TV}} \right)^\star(z) = \imath_{C(c^{-1}\gamma)} \ .$$

On the other hand, by Definition 1, it is easy to check that

$$P_{C(c^{-1}\gamma)} = \mathrm{prox}_{\iota_{C(c^{-1}\gamma)}} \ .$$

Combining the last two equations yields

$$\mathrm{prox}_{\left(c^{-1}\gamma\|.\|_{\mathrm{TV}}\right)^\star} = P_{C(c^{-1}\gamma)}.$$

Identifying $c^{-1}\gamma\|.\|_{\mathrm{TV}}$ with φ and $\left(c^{-1}\gamma\|.\|_{\mathrm{TV}}\right)^\star$ with $\varphi^\star$, equation (45) leads to statement (ii). The proof is complete. □

Note that our argument (45) for the computation of $\mathrm{prox}_{c^{-1}\gamma\|\cdot\|_{\mathrm{TV}}}(u)$ is not used in [18], which instead uses conjugates and bi-conjugates of the objective function.

Remark 3 In view of (44) and (41), one can see that the term between the middle parentheses in (43) admits a simpler form:

$$\mathrm{Id} - \mathrm{prox}_{c^{-1}\gamma\|\cdot\|_{\mathrm{TV}}} = P_{C(c^{-1}\gamma)}.$$

Using (30) along with (43)-(44) we easily find that

$$\mathrm{rprox}_{\gamma\Phi}(x) = \left(\mathrm{Id} - 2W \circ P_{C(c^{-1}\gamma)} \circ \widetilde{W}\right)(x) \ . \tag{46}$$

Calculation of the projection $P_{C(.)}$ in (44) in a discrete setting. In what follows, we work in the discrete setting. We consider that $W \in \mathbb{R}^{M\times N}$ is the analysis matrix associated to a tight frame with $N < M = \#I < \infty$. Note that now $W^* = W^T$ and thus $\widetilde{W} = c^{-1}W^T$.

Next we replace $\mathcal{X}$ by its discrete counterpart,

$$\mathcal{X} = \ell^2(\Omega) \times \ell^2(\Omega) \ \text{ where } \ \#\Omega = N. \tag{47}$$

We denote the discrete gradient by $\ddot{\nabla}$ and consider $\mathrm{Div} : \mathcal{X} \to \ell^2(\Omega)$ the discrete divergence defined by analogy with the continuous setting [2] as the adjoint of the gradient $\mathrm{Div} = -\ddot{\nabla}^*$; see e.g. [18].

Unfortunately, the projection in (44) does not admit an explicit form. The next lemma provides an iterative scheme to compute the proximal points introduced in Lemma 2. In this discrete setting, the set $C(.)$ introduced in (41) admits a simpler expression:

$$C\left(\frac{\gamma}{c}\right) = \left\{\mathrm{Div}(z) \in \ell^2(\Omega) \ \middle|\ z \in B_\infty\left(\frac{\gamma}{c}\right)\right\} \ , \tag{48}$$

where $B_\infty(.)$ is defined according to (40).

[2] More precisely, let $u \in \ell^2(\Omega)$ be of size $m \times n$, $N = mn$. We write

$$(\ddot{\nabla}u)[i,j] = \big(u[i+1,j] - u[i,j],\ u[i,j+1] - u[i,j]\big)$$

with boundary conditions $u[m+1,i] = u[m,i],\ \forall i$ and $u[i,n+1] = u[i,n],\ \forall i$; then for $z \in \mathcal{X}$, we have

$$(\mathrm{Div}(z))[i,j] = \big(z_1[i,j] - z_1[i-1,j]\big) + \big(z_2[i,j] - z_2[i,j-1]\big)$$

along with $z_1[0,i] = z_1[m,i] = z_2[i,0] = z_2[i,n] = 0,\ \forall i$.

Lemma 3 *We adapt all assumptions of Lemma 2 to the discrete setting, as explained above. Given $u \in \ell^2(\Omega)$, consider the forward-backward iteration: $z^{(0)} \in B_\infty(1)$ and, for all $k \in \mathbb{N}$,*

$$z^{(k+1)} = P_{B_\infty(1)}\left(z^{(k)} + \beta_k \ddot{\nabla}\left(\mathrm{Div}(z^{(k)}) - cu/\gamma\right)\right) \quad \textit{where} \quad 0 < \inf_k \beta_k \leq \sup_k \beta_k < 1/4, \tag{49}$$

and, $\forall (i,j) \in \Omega$,

$$P_{B_\infty(1)}(z)[i,j] = \begin{cases} z[i,j] & \textit{if } |z[i,j]| \leq 1; \\ \dfrac{z[i,j]}{|z[i,j]|} & \textit{otherwise .} \end{cases} \tag{50}$$

Then

(i) $(z^{(k)})_{k\in\mathbb{N}}$ converges to a point $\hat{z} \in B_\infty(1)$;

(ii) $\left(u - c^{-1}\gamma \mathrm{Div}(z^{(k)})\right)_{k\in\mathbb{N}}$ converges to $u - c^{-1}\gamma\mathrm{Div}(\hat{z}) = \mathrm{prox}_{c^{-1}\gamma\|\cdot\|_{\mathrm{TV}}}(u)$ at the rate O(1/k).

Proof. Given $u \in \ell^2(\Omega)$, the projection $\hat{w} = P_{C(c^{-1}\gamma)}(u)$, where $C(.)$ is given in (48), is unique and satisfies

$$\begin{aligned} \hat{w} &= \operatorname*{argmin}_{w \in C(c^{-1}\gamma)} \frac{1}{2}\|u-w\|^2 = \operatorname*{argmin}_{w\in\ell^2(\Omega)} \left\{ \frac{1}{2}\left\|\frac{c}{\gamma}u - w\right\|^2 \text{ subject to } w = \mathrm{Div}(z) \text{ for } z \in B_\infty(1)\right\} \\ &\Updownarrow \\ \hat{w} &= \mathrm{Div}(\hat{z}) \text{ where } \hat{z} = \operatorname*{argmin}_{z\in B_\infty(1)} \frac{1}{2}\left\|\frac{c}{\gamma}u - \mathrm{Div}(z)\right\|^2 , \end{aligned} \tag{51}$$

where $\|.\|$ denotes the Euclidian norm on $\ell^2(\Omega)$. This problem can be solved using a projected gradient method (which is a special instance of the forward-backward splitting scheme) whose iteration is given by (49). This iteration converges to a minimizer of (51)—see [25, Corollary 6.5]— provided that the stepsize $\beta_k > 0$ satisfies $\sup_k \beta_k < 2/\delta^2$, where δ is the spectral norm of the Div operator. It is easy to check that $\delta^2 \leq 8$—see e.g. [18]. Hence statement (i).

Next we focus on statement (ii). Set

$$\omega^{(k)} = c\gamma^{-1}u - \mathrm{Div}(z^{(k)}), \forall k \in \mathbb{N} \text{ and } \hat{\omega} = c\gamma^{-1}u - \mathrm{Div}(\hat{z}).$$

Let J be the dual objective given in (51), namely

$$J = H \circ \mathrm{Div} + G, \tag{52}$$

with

$$H : w \mapsto \frac{1}{2}\left\|\frac{c}{\gamma}u - w\right\|^2 \quad \text{and} \quad G = \imath_{B_\infty(1)}.$$

Let $\mathsf{D}H$ denote the the usual derivative of H. For an optimal solution $\hat{z} \in B_\infty(1)$, we define the Bregman-like distance as the functional

$$\mathsf{B}(w) = G(w) - G(\hat{z}) + \left\langle -\ddot{\nabla}(\mathsf{D}H)(\mathrm{Div}(\hat{z})), w - \hat{z}\right\rangle_{\mathcal{X}} , \ \forall\ w \in \mathcal{X} , \tag{53}$$

It is obvious that $\mathsf{B}(\hat{z}) = 0$. We also have $\mathsf{B}(w) \geq 0$, $\forall w \in \mathcal{X}$. This is checked by noting that the minimality condition corresponding to (52) is equivalent to $-\ddot{\nabla}(\mathsf{D}H)(\mathrm{Div}(\hat{z})) \in \partial G(\hat{z})$. Applying the subgradient inequality to G proves the non-negativity claim.

As $H \circ \mathrm{Div}$ is differentiable, the Taylor distance is the remainder of the Taylor expansion of $H \circ \mathrm{Div}$ near $\hat{z}$

$$\mathsf{T}(w) = H(\mathrm{Div}(w)) - H(\mathrm{Div}(\hat{z})) - \left\langle -\ddot{\nabla}(\mathsf{D}H)(\mathrm{Div}(\hat{z})), w - \hat{z} \right\rangle_{\mathcal{X}} \ , \ \forall \ w \in \mathcal{X} \ . \tag{54}$$

Reminding that $\mathrm{Div} = -\ddot{\nabla}^*$ and that H is convex shows that $\mathsf{T}(w) \geq 0$, $\forall w \in \mathcal{X}$, along with $\mathsf{T}(\hat{z}) = 0$.

It is easy to verify that

$$\mathsf{B}(z^{(k)}) + \mathsf{T}(z^{(k)}) = J(z^{(k)}) - J(\hat{z}) \ , \forall k \in \mathbb{N} \ . \tag{55}$$

Using [56] or [48, Theorem 4], the convergence rate over J satisfies

$$J(z^{(k)}) - J(\hat{z}) \quad \leq \quad \frac{2\delta^2 R^2}{k+2} \ , \quad \forall \ k \in \mathbb{N} \ , \tag{56}$$

where $R \stackrel{\text{def}}{=} \max\left\{\|z - \hat{z}\| : J(z) \leq J(z^{(0)})\right\}$ for $z^{(0)} \in B_\infty(1)$. R is obviously finite since $B_\infty(1)$ is bounded. Using (55), along with the facts that $B(z^{(k)}) \geq 0$ and that $\delta^2 \leq 8$ leads to

$$\mathsf{T}(z^{(k)}) \leq J(z^{(k)}) - J(\hat{z}) \leq \frac{16R^2}{k+2} \ , \quad \forall \ k \in \mathbb{N} \ . \tag{57}$$

On the other hand, by the definition of H, which is a quadratic (strongly convex of modulus 1), we have

$$\begin{aligned} \mathsf{T}(z^{(k)}) &= H(\mathrm{Div}(z^{(k)})) - H(\mathrm{Div}(\hat{z})) - \left\langle -\ddot{\nabla}(\mathsf{D}H)(\mathrm{Div}(\hat{z})), z^{(k)} - \hat{z} \right\rangle_{\mathcal{X}} \\ &= H(\mathrm{Div}(z^{(k)})) - H(\mathrm{Div}(\hat{z})) - \left\langle (\mathsf{D}H)(\mathrm{Div}(\hat{z})), \mathrm{Div}(z^{(k)}) - \mathrm{Div}(\hat{z}) \right\rangle \\ &= \frac{1}{2}\left\|\mathrm{Div}(z^{(k)}) - \mathrm{Div}(\hat{z})\right\|^2 = \frac{1}{2}\left\|\omega^{(k)} - \hat{\omega}\right\|^2 \ . \end{aligned} \tag{58}$$

Piecing together (58) and (57), we obtain

$$\left\|\omega^{(k)} - \hat{\omega}\right\|^2 = 2\mathsf{T}(z^{(k)}) \leq \frac{32R^2}{k+2} \ , \quad \forall \ k \in \mathbb{N} \ , \tag{59}$$

This completes the proof.

□

Note that computing $\mathrm{prox}_{\|\cdot\|_{\mathrm{TV}}}$ amounts to solving a discrete L^2-TV-denoising (the criterion given in (10)).

The forward-backward splitting-based iteration proposed in (49) to compute the proximity operator of the TV-norm is different from the projection algorithm of [18]. A similar iteration was suggested in [19] without a proof. The forward-backward splitting allows to derive a sharper upper-bound on the stepsize β_k than the one proposed in [18]—actually twice as large. What is more, our proof is simpler than the

one in [18] since it uses well known properties of proximity operators, and we have a convergence rate on the iterates.

Our iteration to solve this problem is one possibility among others. While this paper was submitted, our attention was drawn to an independent work of [10] who, using a different framework, derive an iteration similar to (49) to solve the L^2-TV-denoising. Another parallel work of [74] propose an application of gradient projection to solving the dual problem (51), and the authors of [71] applied the multi-step Nesterov scheme to (51). See also [12] for yet another multi-step iteration to solve (51). We are of course aware of max-flow/min-cut type algorithms, for instance the one in [16]. We have compared our whole denoising procedure using our implementation of $\mathrm{prox}_{\|\cdot\|_{\mathrm{TV}}}$ and the max-flow based implementation that we adapted from the code available at [17]. We obtained similar results, although the max-flow-based algorithm was faster, mainly because it uses the ℓ^1 approximation of the discrete gradient, namely $\left\|(\ddot{\nabla}u)[i,j]\right\|_1 = \left|u[i+1,j]-u[i,j]\right| + \left|u[i,j+1]-[i,j]\right|$. Let us remind that this approximation for the discrete gradient does not inherit the rotational invariance property of the L^2 norm of the usual gradient; we observed that the quality of the resultant restorations is slightly reduced, compared to those involving the L^2 norm of the discrete gradient.

3.3 Comments on the Douglas-Rachford scheme for F_y

The bottleneck of the minimization algorithm is in the computation of the proximity-operator of the TV-norm (Lemma 2). However, when inserted in the whole numerical scheme, this is not a real drawback as we explain it below. A crucial property of the Douglas-Rachford splitting scheme (33) is its robustness to numerical errors that may occur when computing the proximity operators prox_{Ψ} and prox_{Φ}, see [25]. We have deliberately omitted this property in (33) for the sake of simplicity. This robustness property has important consequences: e.g. it allows us to run the forward-backward sub-recursion (49) only a few iterations to compute an approximate of the TV-norm proximity operator in the inner iterations, and the Douglas-Rachford is still guaranteed to converge provided that these numerical errors are under control. More precisely, let $a_k \in \ell^2(I)$ be an error term that models the inexact computation of $\mathrm{prox}_{\gamma\Phi}$ in (43), as the latter is obtained through (49). If the sequence of error terms $(a_k)_{k\in\mathbb{N}}$ and step-sizes $(\mu_k)_{k\in\mathbb{N}}$ defined in Theorem 3 obey $\sum_{k\in\mathbb{N}} \mu_k \|a_k\| < +\infty$, then the Douglas-Rachford algorithm (33) converges [25, Corollary 6.2]. In our case, noting that the convergence rate of Lemma 3 yields $\|a_k\| = O(1/k)$, one can easily derive a rule on the number of inner iterations at each outer iteration k such that the summability condition is verified.

4 Bias correction to recover the sought-after image

Recall from (5) that $u_o = \log S_o$ and set $\hat{u} = \widetilde{W}\hat{x}$ as the estimator of u_o, where $\hat{x}$ is a minimizer of F_y obtained from the Douglas-Rachford iteration. Unfortunately, the estimator $\hat{u}$ is prone to bias, i.e. $\mathbb{E}[\hat{u}] = u_o - b_{\hat{u}}$. A problem that classically arises in statistical estimation is how to correct such a bias. More importantly is how this bias affects the estimate after applying the inverse transformation, here the exponential. Our goal is then to ensure that for the estimate $\hat{S}$ of the image satisfies $\mathbb{E}\left[\hat{S}\right] = S_o$.

To this end we need the expectation and the variance of the log-noise n in (5)-(6). One can prove

that

$$\begin{aligned} \mathbb{E}[n] &= \boldsymbol{\psi}_0(K) - \log K \ , \quad (60) \\ \mathrm{Var}[n] &= \boldsymbol{\psi}_1(K), \quad (61) \end{aligned}$$

where

$$\boldsymbol{\psi}_k(z) = \left(\frac{d}{dz}\right)^{k+1} \log \Gamma(z) \quad (62)$$

is the polygamma function [1].

Expanding $\exp \hat{u}$ in the neighborhood of $\mathbb{E}[\hat{u}]$, we have

$$\exp \hat{u} = \exp\left(\mathbb{E}[\hat{u}]\right) \left(1 + (\hat{u} - \mathbb{E}[\hat{u}]) + \frac{(\hat{u} - \mathbb{E}[\hat{u}])^2}{2} + R_2\right),$$

and therefore

$$\begin{aligned} \mathbb{E}[\exp \hat{u}] &= \exp\left(\mathbb{E}[\hat{u}]\right)(1 + \mathrm{Var}[\hat{u}]/2 + \mathbb{E}[R_2]) \\ &= S_o \exp\left(-b_{\hat{u}}\right)(1 + \mathrm{Var}[\hat{u}]/2 + \mathbb{E}[R_2]) \ , \quad (63) \end{aligned}$$

where R_2 is the Lagrange remainder in the Taylor series. One can observe that the posterior distribution of $\hat{u}$ is nearly symmetric, in which case $\mathbb{E}[R_2] \approx 0$. That is, $b_{\hat{u}} \approx \log(1+\mathrm{Var}[\hat{u}]/2)$ to ensure unbiasedness. Consequently, finite sample (nearly) unbiased estimates of u_o and S_o are respectively $\hat{u}+\log(1+\mathrm{Var}[\hat{u}]/2)$, and $\exp(\hat{u})(1 + \mathrm{Var}[\hat{u}]/2)$. $\mathrm{Var}[\hat{u}]$ can be reasonably estimated by $\boldsymbol{\psi}_1(K)$, the variance of the noise n in (5) being given in (61). Thus, given the restored log-image $\hat{u}$, our restored image reads:

$$\hat{S} = \exp(\hat{u})\,(1 + \boldsymbol{\psi}_1(K)/2) \ . \quad (64)$$

The authors of [72] propose a direct estimate of the bias $b_{\hat{u}}$ using the obvious argument that the noise n in the log-transformed image has a non-zero mean $\boldsymbol{\psi}_0(K) - \log K$. A quick study shows that the functions $(1 + \boldsymbol{\psi}_1(K)/2)$ and $\exp(\log K - \boldsymbol{\psi}_0(K))$ are very close for K reasonably large.

We should emphasize that the bias correction approach we propose can be used in a more general setting.

5 Full algorithm to suppress multiplicative noise

Now, piecing together Lemma 1, Lemma 2 and Theorem 3, we arrive at the multiplicative noise removal algorithm:

Task: Denoise an image S contaminated with multiplicative noise according to (3).

Parameters: The observed noisy image S, number of iterations N_{DR} (Douglas-Rachford outer iterations) and N_{FB} (Forward-Backward inner iterations), stepsizes $\mu_k \in (0, 2)$, $0 < \beta_s < 1/4$ and $\gamma > 0$, tight-frame transform W and initial threshold T (e.g. $T = 2\sqrt{\psi_1(K)}$), regularization parameters $\lambda_{0,1}$ associated to the sets $I_{0,1}$.

Specific operators:

- Soft-thresholding $\mathcal{T}_{\mathrm{S}}{}^{\gamma\lambda_i}$ in (36).

- Projector $P_{B_\infty(1)}$ in (50).
- The discrete gradient $\ddot{\nabla}$ and divergence Div.
- $\psi_1(\cdot)$ defined according to (62) (built-in Matlab function, otherwise see [57]).

Initialization:

- Compute $v = \log S$ and transform coefficients $y = Wv$. Hard-threshold y at T to get $y_{\mathcal{T}_\mathrm{H}}$. Choose an initial $x^{(0)}$.

Main iteration:
For $k = 0$ **to** N_{DR},

(1) Inverse curvelet transform of $x^{(k)}$ according to $u^{(k)} = \widetilde{W} x^{(k)}$.

(2) Initialize $z^{(0)}$; **For** $s = 0$ **to** $N_{\mathrm{FB}} - 1$

$$z^{(s+1)} = P_{B^1_\infty(\mathcal{X})}\left(z^{(s)} + \beta_s \ddot{\nabla}\left(\mathrm{Div}(z^{(s)}) - \tfrac{c}{\gamma} u^{(k)}\right)\right).$$

(3) Set $z^{(k)} = z^{(N_{\mathrm{FB}})}$.

(4) Compute $w^{(k)} = c^{-1}\gamma\ \mathrm{Div}(z^{(k)})$.

(5) Forward curvelet transform: $\alpha^{(k)} = W w^{(k)}$.

(6) From (46) compute $r^{(k)} = \mathrm{rprox}_{\gamma\Phi}(x^{(k)}) = x^{(k)} - 2\alpha^{(k)}$.

(7) By (39) compute $q^{(k)} = \left(\mathrm{rprox}_{\gamma\Psi} \circ \mathrm{rprox}_{\gamma\Phi}\right)(x^{(k)}) = 2\left(y_{\mathcal{T}_\mathrm{H}}[i] + \mathcal{T}_\mathrm{S}{}^{\gamma\lambda_i}\left(r^{(k)}[i] - y_{\mathcal{T}_\mathrm{H}}[i]\right)\right)_{i\in I} - r^{(k)}$.

(8) Update $x^{(k+1)}$ using (33): $x^{(k+1)} = \left(1 - \dfrac{\mu_k}{2}\right) x^{(k)} + \dfrac{\mu_k}{2} q^{(k)}$.

End main iteration
Output: Using (34) and (43), get the denoised image $\hat{S} = \exp\left(\widetilde{W}(x^{(N_{\mathrm{DR}})} - \alpha^{(N_{\mathrm{DR}})})\right)(1 + \psi_1(K)/2)$.

Remark 4 (Computation load) The bulk of computation of our denoising algorithm is invested in applying W and its pseudo-inverse $\widetilde{W}$. These operators are of course never constructed explicitly, rather they are implemented as fast implicit analysis and synthesis operators. Each application of W or $\widetilde{W}$ cost $\mathcal{O}(N \log N)$ for the second generation curvelet transform of an N-pixel image [14]. If we define N_{DR} and N_{FB} as the number of iterations in the Douglas-Rachford algorithm and the forward-backward sub-iteration, the computational complexity of the denoising algorithm is of order $N_{\mathrm{DR}} N_{\mathrm{FB}} 2N \log N$ operations.

6 Experiments

In all experiments carried out in this paper, our algorithm was run using second-generation curvelet tight frame along with the following set of parameters: $\forall t, \mu_k \equiv 1$, $\beta_s = 0.24$, $\gamma = 10$ and $N_{\mathrm{DR}} = 50$. For comparison purposes, some very recent multiplicative noise removal algorithms from the literature are considered. We compare our method with the most recent successful algorithms, namely:

- **BS** Algorithm proposed in [22]: this is a Stein-Block thresholding method in the curvelet domain, applied on the log transformed image. It is a sophisticated shrinkage-based denoiser that thresholds the coefficients by blocks rather than individually, and has been shown to be nearly minimax over a large class of images in presence of additive bounded noise (not necessarily Gaussian nor independent). This algorithm has one threshold parameter $\mathfrak{T}$, and in all our experiments, we set it to the theoretical value $\underline{\mathfrak{T} = 4.50524}$ devised in [22].

- **AA** Algorithm proposed in [8]: it is sketched in (8); the algorithm was applied using $\underline{1000 \text{ iterations}}$ and stepsize $\underline{dt = 0.1}$, as recommended by the authors;

- **SO** Algorithm proposed in [62]: the first algorithm proposed in [62], see (10). As recommended in [62] the stopping rule is to reach k^* such that $k^* = \max\{k \in \mathbb{N} : \mathrm{Var}\,[u_k - u_o] \geq \mathrm{Var}\,[n] = \psi_1(K)\}$ where u_o is the underlying log-image and n the relevant noise; see (61) for the variance. We systematically used stepsize $\underline{dt = 0.001}$ (except for Fig. 3 where $dt = 0.0005$).

The denoising algorithms were tested on three images: Cameraman ($N = 256^2$ pixels), an image of Fields ($N = 512^2$ pixels) and an aerial picture of the French city Nîmes ($N = 512^2$ pixels). All images were normalized so that their gray-scale is in the range $[1, 256]$. For each image, a noisy observation is generated by multiplying the original image by a realization of noise according to the model in (3)-(4) with the choice $\mu = 1$ and $K \in \{1, 4, 10\}$. For a N-pixel noise-free image S_o and its denoised version by any algorithm $\hat{S}$, the denoising performance is measured in terms of peak signal to noise ratio (PSNR) in decibels (dB)

$$\mathrm{PSNR} = 10 \log_{10} \frac{N \left| \max S_o - \min S_o \right|^2}{\left\| \hat{S} - S_o \right\|_2^2} \text{ dB} ,$$

where $\left| \max S_o - \min S_o \right|$ gives the gray-scale range of the original image, and mean absolute-deviation error MAE

$$\mathrm{MAE} = \frac{1}{N} \left\| \hat{S} - S_o \right\|_1 .$$

6.1 Qualitative results

The results are depicted in Fig. 3-6 for the Fields image, Fig. 7-9 for the Nîmes picture, and Fig. 10-12 for Cameraman. Our denoiser does a good job at restoring faint geometrical structures of the images even for low values of K, see for instance the results on Nîmes and Fields for $K = 1$ and $K = 4$. As expected, our hybrid method is less prone to staircasing artifacts and takes benefit from the curvelet transform for capturing efficiently the geometrical content of the images. Our algorithm performs among the best and even outperforms its competitors most of the time both visually and quantitatively as revealed by the PSNR and MAE values. Note also that a systematic behavior of AA and SO methods for low values of K is their tendency to lose some important details and the persistence of a low-frequency ghost as it can be seen on the residual images on Fig. 4. For SO method, the number of iterations necessary to satisfy the stopping rule rapidly increases when K decreases.

6.2 Quantitative results

The above visual results were confirmed by Monte Carlo simulations where, for each tested image and each value of $K \in \{1, 4, 10\}$, ten noisy realizations were generated. Then the compared algorithms were applied to the same noisy versions. The output PSNR and MAE were averaged over the ten denoised realizations.

For fair comparison, the parameters of SO and AA were tweaked manually to reach their best performance level. For SO, $\underline{\alpha = 0.25}$ was recommended by the authors. Their values are summarized in Table 1. Notice that the parameters of our method are very stable with respect to the image, except for Cameraman with small K. Curvelet based methods are indeed quite inadequate to this image since the thresholding step generates Gibbs-like oscillations and outliers that are strongly visible on the nearly uniform background (see the image restored by BS method). In order to remove these artifacts, we used different values for T, λ_0 and mostly λ_1. Despite this drawback, our method yields quite good results on Cameraman.

The denoising performance results are tabulated in Table 2 where the best PSNR and MAE value is shown in boldface. By inspection of this table, the PSNR improvement brought by our approach can be quite high particularly for $K = 1$ (see e.g. the Fields image) and the visual resolution is quite respectable. This is an important achievement since in practice K has a small value, usually 1, rarely above 4. This improvement becomes less salient as K increases which is intuitively expected. But even for $K = 10$, the PSNR of our algorithm can be higher by ~ 0.45dB to more than 10dB compared to AA, SO and BS methods, depending on the image.

Fields 512×512

Algorithm	Parameters		
	$K = 1$	$K = 4$	$K = 10$
Ours	$T = 2\sqrt{\psi_1(K)}, \lambda_0 = 1.2, \lambda_1 = 10$	$T = 2\sqrt{\psi_1(K)}, \lambda_0 = 1.3, \lambda_1 = 10$	$T = 2\sqrt{\psi_1(K)}, \lambda_0 = 1.3, \lambda_1 = 10$
AA	$\rho = 125$	$\rho = 400$	$\rho = 480$
SO	$\rho = 0.05$, $\alpha = 0.25$	$\rho = 0.1$, $\alpha = 0.25$	$\rho = 0.8$, $\alpha = 0.25$

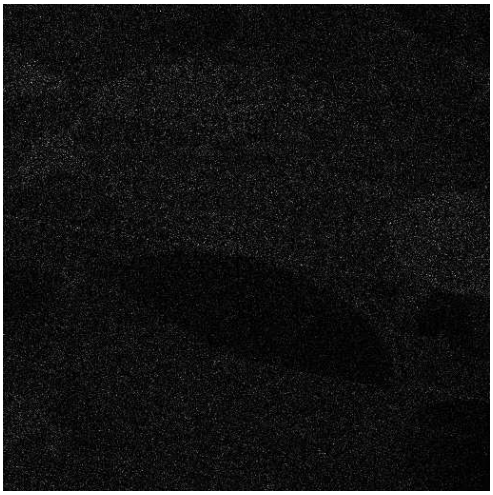
Nîmes 512×512

Algorithm	Parameters		
	$K = 1$	$K = 4$	$K = 10$
Ours	$T = 2\sqrt{\psi_1(K)}, \lambda_0 = 1.5, \lambda_1 = 10$	$T = 2\sqrt{\psi_1(K)}, \lambda_0 = 1.5, \lambda_1 = 10$	$T = 2\sqrt{\psi_1(K)}, \lambda_0 = 1.3, \lambda_1 = 10$
AA	$\rho = 60$	$\rho = 120$	$\rho = 130$
SO	$\rho = 0.05$, $\alpha = 0.25$	$\rho = 0.3$, $\alpha = 0.25$	$\rho = 1.2$, $\alpha = 0.25$

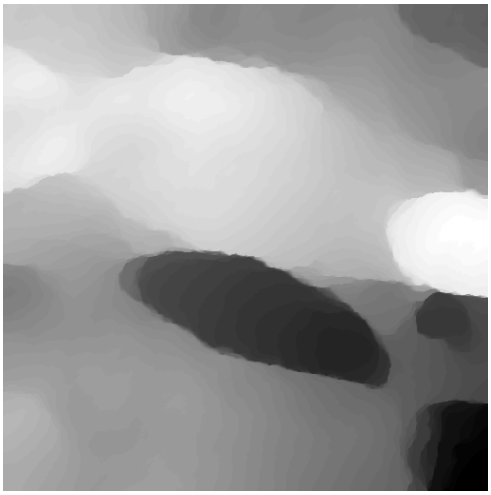
Cameraman 256×256

Algorithm	Parameters		
	$K = 1$	$K = 4$	$K = 10$
Ours	$T = 2.6\sqrt{\psi_1(K)}, \lambda_0 = 1.8, \lambda_1 = 5.7$	$T = 2.5\sqrt{\psi_1(K)}, \lambda_0 = 1.8, \lambda_1 = 5.7$	$T = 2.1\sqrt{\psi_1(K)}, \lambda_0 = 1.3, \lambda_1 = 10$
AA	$\rho = 120$	$\rho = 240$	$\rho = 390$
SO	$\rho = 0.04$, $\alpha = 0.25$	$\rho = 0.1$, $\alpha = 0.25$	$\rho = 1$, $\alpha = 0.25$

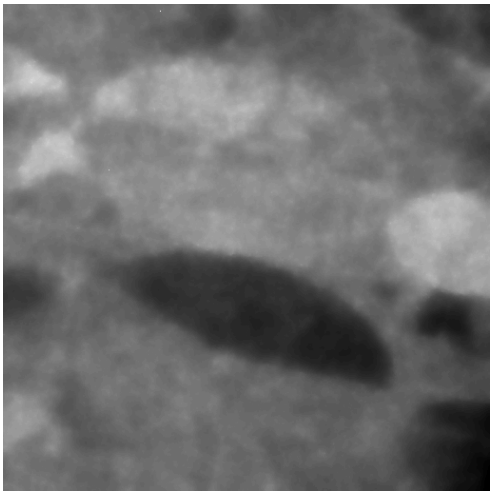
Table 1: Parameters used in the comparison study of Table 2.



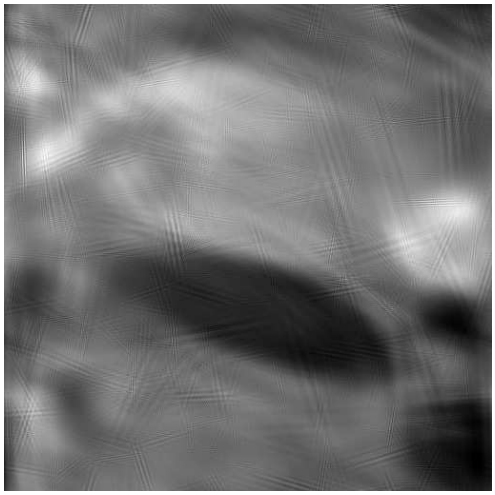
Noisy: $K = 1$ in (3)



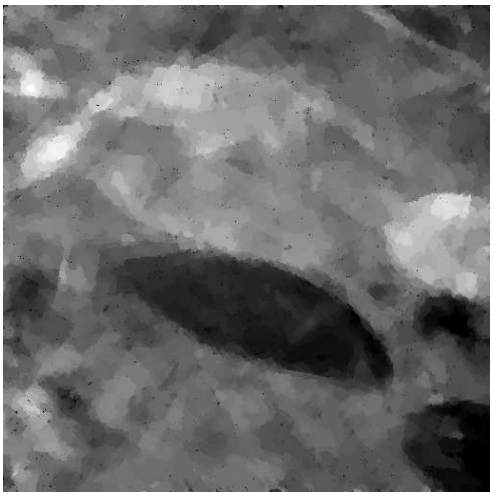
SO: PSNR=9.59, MAE=196



AA: PSNR=15.74, MAE=76.66



BS: PSNR=22.52, MAE=35.22

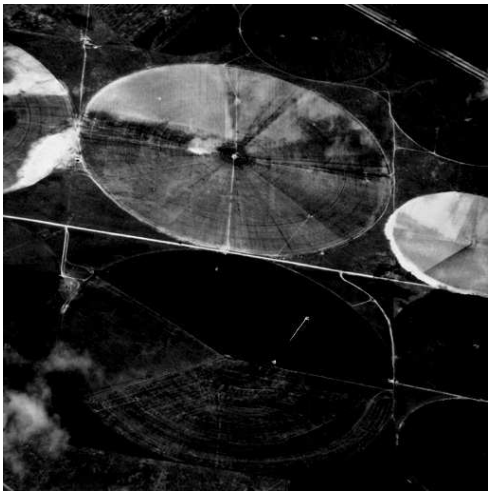


Ours: PSNR=22.89, MAE=33.67

Original (512×512)

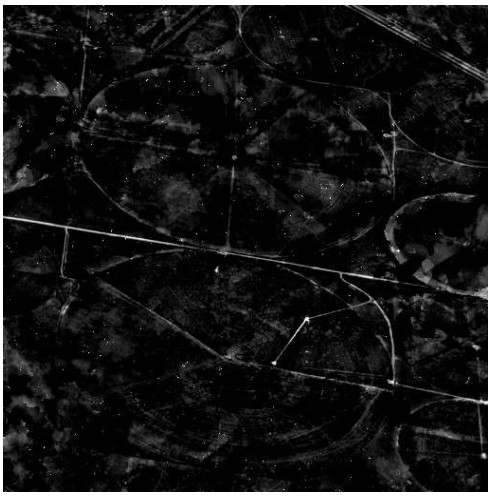
Figure 3: Fields (512 × 512) for $K = 1$ in (3). Restorations using different methods. Parameters: SO algorithm for $\rho = 0.05$, $\alpha = 0.25$ and $dt = 0.0005$; AA algorithm for $\rho = 125$; our algorithm $T = 2\sqrt{\psi_1(K)}, \lambda_0 = 1.2, \lambda_1 = 10$.

SO: residual, $K = 1$



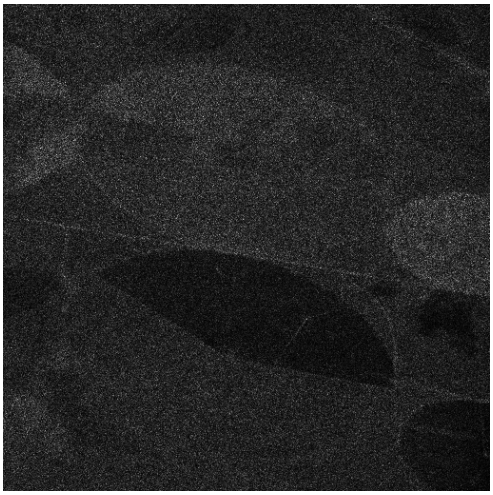
AA: residual, $K = 1$

BS: residual, $K = 1$

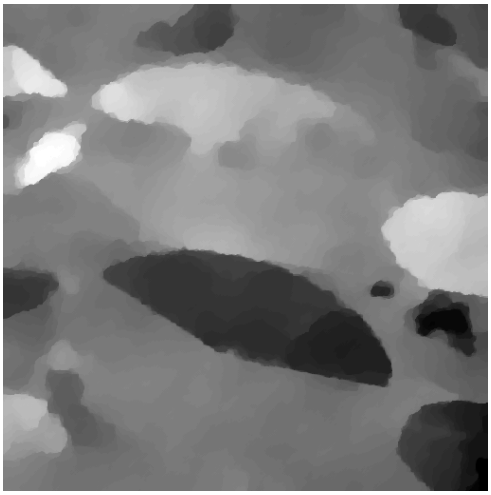


Ours: residual, $K = 1$

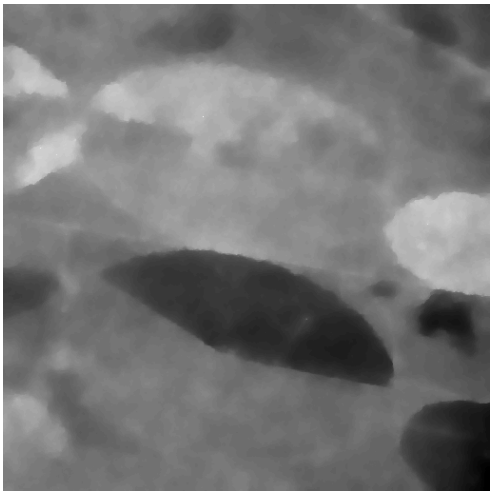
Figure 4: Residuals $(S_o - \hat{S})$ for the restorations in Fig. 3 $(K = 1)$.



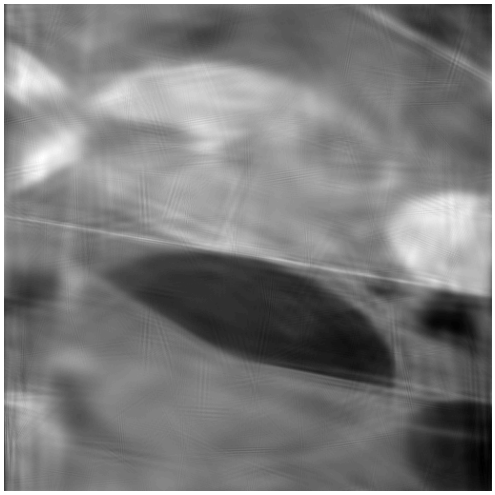
Noisy: $K = 4$ in (3)



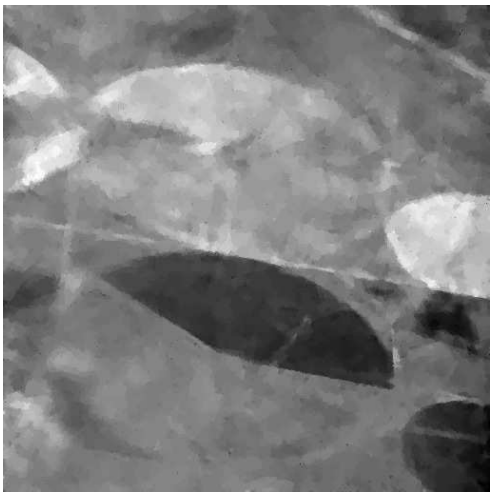
SO: PSNR=19.78, MAE=55.42



AA: PSNR=16.83, MAE=67.74



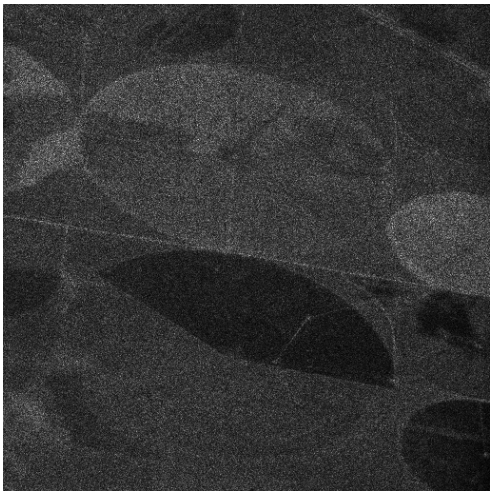
BS: PSNR=25.53, MAE=23.75



Ours: PSNR=26.32, MAE=21.92

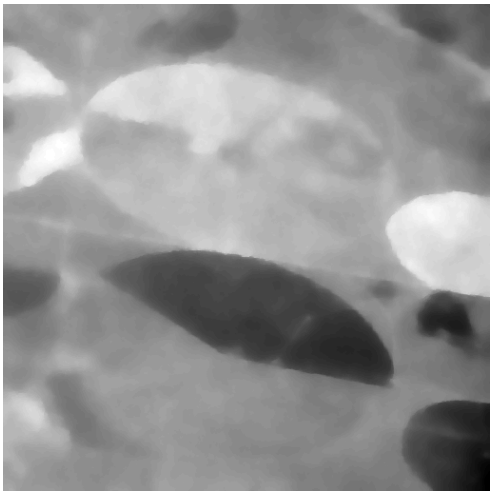
Original (512×512)

Figure 5: Fields (512 × 512) for $K = 4$ in (3). Restorations using different methods. Parameters: SO algorithm for $\rho = 0.1$, $\alpha = 0.25$ and $dt = 0.001$; AA algorithm for $\rho = 400$; our algorithm $T = 2\sqrt{\psi_1(K)}, \lambda_0 = 1.3, \lambda_1 = 10$.

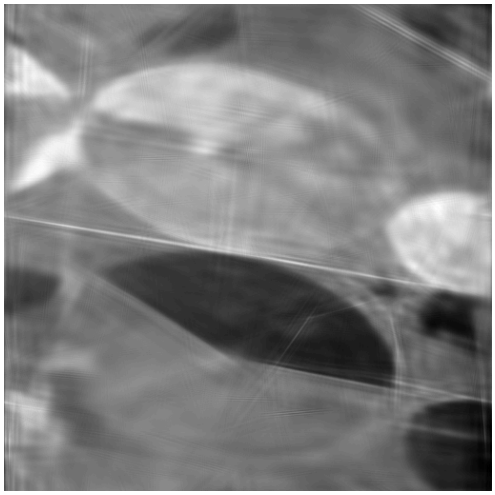


Noisy: $K = 10$ in (3)

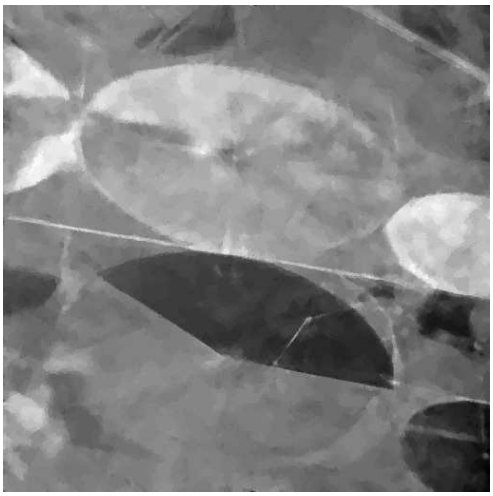
SO: PSNR=25.36, MAE=25.14



AA: PSNR=17.13, MAE=65.40



BS: PSNR=27.24, MAE=19.61

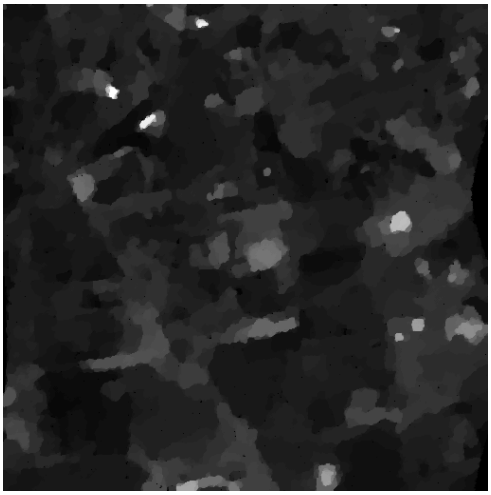


Ours: PSNR=28.04, MAE=18.19

Original (512×512)

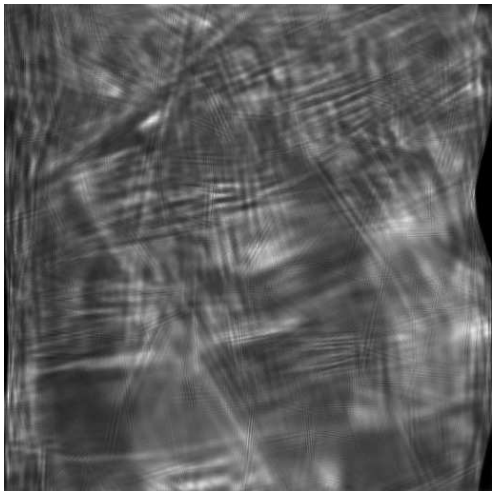
Figure 6: Fields (512×512) for $K = 10$ in (3). Restorations using different methods. Parameters: SO algorithm for $\rho = 0.8$, $\alpha = 0.25$ and $dt = 0.001$; AA algorithm for $\rho = 480$; our algorithm $T = 2\sqrt{\psi_1(K)}, \lambda_0 = 1.3, \lambda_1 = 10$.

Noisy: $K = 1$ in (3)

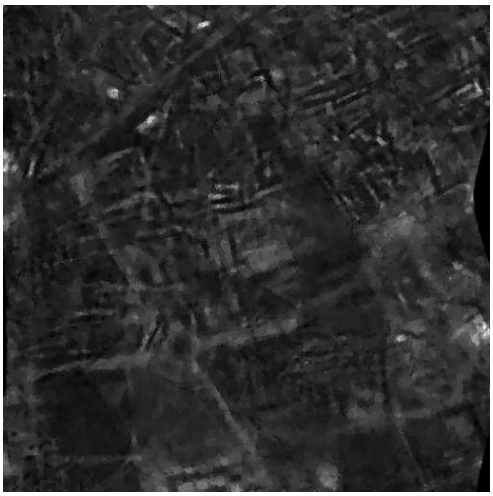


SO: PSNR=18.39, MAE=24.08

AA: PSNR=22.18, MAE=13.71



BS: PSNR=22.25, MAE=13.96



Ours: PSNR=22.64, MAE=13.39

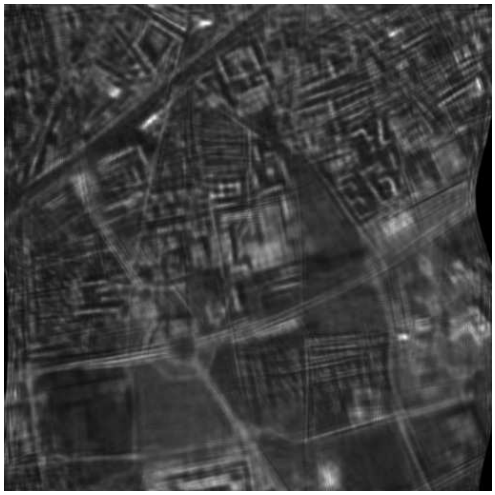
Original (512×512)

Figure 7: Aerial image of the French city of Nîmes (512 × 512) for $K = 1$ in (3). Restorations using different methods. Parameters: SO algorithm for $\rho = 0.05$, $\alpha = 0.25$ and $dt = 0.001$; AA algorithm for $\rho = 60$; our algorithm $T = 2\sqrt{\psi_1(K)}, \lambda_0 = 1.5, \lambda_1 = 10$.

Noisy: $K = 4$ in (3)

SO: PSNR=24.40, MAE=10.76

AA: PSNR=24.55, MAE=10.06



BS: PSNR=24.92, MAE=9.87

Ours: PSNR=25.84, MAE=9.09

Original (512×512)

Figure 8: Aerial image of the town of Nîmes (512 × 512) for $K = 4$ in (3). Restorations using different methods. Parameters: SO algorithm for $\rho = 0.3$, $\alpha = 0.25$ and $dt = 0.001$; AA algorithm for $\rho = 120$; our algorithm $T = 2\sqrt{\psi_1(K)}, \lambda_0 = 1.5, \lambda_1 = 10$.

Noisy: $K = 10$ in (3)

SO: PSNR=27.32, MAE=7.41

AA: PSNR=25.26, MAE=8.83

BS: PSNR=27.0, MAE=7.69

Ours: PSNR=27.80, MAE=7.21

Original (512×512)

Figure 9: Aerial image of Nîmes (512 × 512) for $K = 10$ in (3). Restorations using different methods. Parameters: SO algorithm for $\rho = 1.2$, $\alpha = 0.25$ and $dt = 0.001$; AA algorithm for $\rho = 130$; our algorithm $T = 2\sqrt{\psi_1(K)}, \lambda_0 = 1.3, \lambda_1 = 10$.

Noisy: $K = 1$ in (3)

SO: PSNR=12.74, MAE=48.59

AA: PSNR=18.76, MAE=22.36

BS: PSNR=19.39, MAE=17.49

Ours: PSNR=19.82, MAE=16.33

Original (256×256)

Figure 10: Cameraman (256×256) for $K = 1$ in (3). Restorations using different methods. Parameters: SO algorithm for $\rho = 0.04$, $\alpha = 0.25$ and $dt = 0.001$; AA algorithm for $\rho = 125$; our algorithm $T = 2.6\sqrt{\psi_1(K)}, \lambda_0 = 1.8, \lambda_1 = 5.7$.

Noisy: $K = 4$ in (3)

SO: PSNR=20.12, MAE=18.64

AA: PSNR=21.97, MAE=14.26

BS: PSNR=22.36, MAE=12.14

Ours: PSNR=22.98, MAE=10.61

Original (256×256)

Figure 11: Cameraman (256×256) for $K = 4$ in (3). Restorations using different methods. Parameters: SO algorithm for $\rho = 0.1$, $\alpha = 0.25$ and $dt = 0.001$; AA algorithm for $\rho = 125$; our algorithm $T = 2.5\sqrt{\psi_1(K)}, \lambda_0 = 1.8, \lambda_1 = 5.7$.

Noisy: $K = 10$ in (3)

SO: PSNR=25.64, MAE=7.42

AA: PSNR=24.48, MAE=9.9

BS: PSNR=24.43, MAE=9.44

Ours: PSNR=26.08, MAE=7.4

Original (256×256)

Figure 12: Cameraman (256×256) for $K = 10$ in (3). Restorations using different methods. Parameters: SO algorithm for $\rho = 1$, $\alpha = 0.25$ and $dt = 0.001$; AA algorithm for $\rho = 125$; our algorithm $T = 2.1\sqrt{\psi_1(K)}, \lambda_0 = 1.3, \lambda_1 = 10$.

Fields 512×512

PSNR

K	1	4	10
Ours	**23.13 (0.14)**	**26.31 (0.08)**	**27.99 (0.05)**
BS	22.60 (0.10)	25.50 (0.06)	27.21 (0.05)
AA	15.75 (0.004)	16.82 (0.004)	17.14 (0.004)
SO	10.11 (0.21)	19.70 (0.23)	25.33 (0.08)

MAE

K	1	4	10
Ours	**32.67 (0.68)**	**22.00 (0.18)**	**18.24 (0.10)**
BS	35.00 (0.58)	23.95 (0.22)	19.66 (0.12)
AA	76.64 (0.04)	67.77 (0.05)	65.36 (0.04)
SO	189.24 (2.96)	55.80 (1.45)	25.26 (0.22)

Nîmes 512×512

PSNR

K	1	4	10
Ours	**22.66 (0.09)**	**25.86 (0.03)**	**27.78 (0.03)**
BS	22.18 (0.03)	24.90 (0.03)	26.95 (0.02)
AA	22.01 (0.04)	24.33 (0.01)	25.06 (0.02)
SO	18.26 (0.03)	24.25 (0.04)	26.84 (0.03)

MAE

K	1	4	10
Ours	**13.27 (0.07)**	**8.98 (0.03)**	**7.11 (0.02)**
BS	13.88 (0.04)	9.75 (0.03)	7.59 (0.01)
AA	14.05 (0.09)	10.37 (0.02)	9.03 (0.02)
SO	24.54 (0.08)	10.99 (0.04)	7.83 (0.02)

Cameraman 256×256

PSNR

K	1	4	10
Ours	**19.61 (0.12)**	**22.94 (0.07)**	**26.09 (0.10)**
BS	19.22 (0.09)	22.31 (0.07)	24.40 (0.05)
AA	18.65 (0.12)	21.93 (0.09)	24.42 (0.07)
SO	12.62 (0.28)	19.96 (0.12)	25.61 (0.07)

MAE

K	1	4	10
Ours	**16.78 (0.25)**	**10.67 (0.07)**	**7.44 (0.09)**
BS	17.94 (0.25)	12.14 (0.12)	9.51 (0.06)
AA	22.61 (0.34)	14.39 (0.15)	9.99 (0.10)
SO	49.33 (1.56)	19.04 (0.32)	7.43 (0.04)

Table 2: Average PSNR and MAE (over ten noisy realizations) to denoise different images using the SO, AA BS and our algorithm as a function of K. The standard deviation of PSNR and MAE over the ten realizations are in parentheses. The best PSNR and MAE value is shown in boldface. The parameters used for each denoising algorithm are summarized in Table 1.

7 Conclusions

This work proposes quite an original, efficient and fast method for multiplicative noise removal. The latter is a difficult problem that arises in various applications relevant to active imaging system, such as laser imaging, ultrasound imaging, SAR and many others. Multiplicative noise contamination involves inherent difficulties that severely restrict the main restoration algorithms.

The main ingredients of our method are: (1) consider the log-data to restore a log-image; (2) preprocess the log-data using and under-optimal hard-thresholding of its tight frame coefficients; (3) restore the log-image using a hybrid criterion composed of an ℓ^1 data-fitting for the coefficients and a TV regularization in the log-image domain; (4) restore the sought-after image using an exponential transform along with a pertinent bias correction. The resultant algorithm is fast, its consistency and convergence are proved theoretically.

The obtained numerical results are really encouraging since they outperform the most recent methods in this field.

Acknowledgments

The authors acknowledge the support of the French Agence Nationale de la Recherche (ANR), under grants FREEDOM and NatImages (ANR07-JCJC-0048-01 and ANR08-EMER-009). The authors would like to thank J.-F. Aujol for providing us the Matlab code of his algorithm and giving us wise advices on the choice of the parameters. They also thank J. Shi and S. Osher for the fruitful discussions on their approach and the advices on the choice of the parameters, yet again.

References

[1] M. Abramowitz and I. A. Stegun. Handbook of mathematical functions. Dover Publications, New York, 1972.

[2] R. Acar and C. Vogel. Analysis of bounded variation penalty methods for ill-posed problems. IEEE Transactions on Image Processing, 10(6):1217–1229, Dec. 1994.

[3] A. Achim, A. Bezerianos and P. Tsakalides. Novel bayesian multiscale method for speckle removal in medical ultrasound images. IEEE Trans. Med. Imaging, 20(8):772–783, Aug. 2001.

[4] A. Achim, E. Kuruoglu and J. Zerubia. SAR image filtering based on the heavy-tailed rayleigh model. IEEE Transactions on Image Processing, 15(9):2686–2693, Sep. 2006.

[5] A. Achim, P. Tsakalides and A. Bezerianos. SAR image denoising via bayesian wavelet shrinkage based on heavy-tailed modeling. IEEE Trans. Geosci. Remote Sens., 41(8):1773–1784, Aug. 2003.

[6] A. Antoniadis and J. Fan. Regularization of wavelet approximations. Journal of Acoustical Society America, 96(455):939–967, Sep. 2001.

[7] A. Antoniadis, D. Leporini and J.-C. Pesquet. Wavelet thresholding for some classes of non-gaussian noise. Statistica Neerlandica, 56(4):434–453, Dec. 2002.

[8] G. Aubert and J.-F. Aujol. A variational approach to remove multiplicative noise. SIAM Journal on Applied Mathematics, 68(4):925–946, Jan. 2008.

[9] G. Aubert and P. Kornprobst. Mathematical problems in image processing. Springer-Verlag, Berlin, 2 edition, 2006.

[10] J.-F. Aujol. Some first-order algorithms for total variation based image restoration. Journal of Mathematical Imaging and Vision, in press, available online 2009.

[11] M. Belge, M. Kilmer and E. Miller. Wavelet domain image restoration with adaptive edge-preserving regularization. IEEE Transactions on Image Processing, 9(4):597–608, Apr. 2000.

[12] A. Beck and M. Teboulle A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems. SIAM Journal on Imaging Sciences, 2(1):183–202, Apr. 2009.

[13] Y. Bobichon and A. Bijaoui. Regularized multiresolution methods for astronomical image enhancement. Exper. Astron., (7):239–255, 1997.

[14] E. J. Candès, D. Donoho and L. Ying. Fast discrete curvelet transforms. SIAM Multiscale Model. Simul., 5(3):861–899, Jan. . 2005.

[15] E. J. Candès and F. Guo. New multiscale transforms, minimum total variation synthesis. Applications to edge-preserving image reconstruction. Signal Processing, 82, Mar. 2002.

[16] A. Chambolle and J. Darbon. On total variation minimization and surface evolution using parametric maximum flows. Technical report, CMAP-04, Ecole Polytechnique, France, 2004.

[17] `http://www.cmap.polytechnique.fr/~antonin/software/`.

[18] A. Chambolle. An algorithm for total variation minimization and application. Journal of Mathematical Imaging and Vision, 20(1), Jan. -Mar. 2004.

[19] A. Chambolle. Total variation minimization and a class of binary MRF models. LNCS 3757:136–152, 2005.

[20] T.F. Chan and H.M. Zhou. Total variation improved wavelet thresholding in image compression. In Proceedings of the IEEE International Conference on Image Processing, volume 2, pages 391–394. IEEE, 2000.

[21] P. Charbonnier, L. Blanc-Féraud, G. Aubert and M. Barlaud. Deterministic edge-preserving regularization in computed imaging. IEEE Transactions on Image Processing, 6(2):298–311, Feb. 1997.

[22] C. Chesneau, M.J. Fadili and J.-L. Starck. Stein block thresholding for image denoising. Applied and Computational Harmonic Analysis, 28(1):67–88, 2010.

[23] R. R. Coifman and D. Donoho. Translation-invariant de-noising. In Wavelets and Statistics; Antoniadis, A.; Oppenheim, G., Eds.; Department of Statistics, 1995.

[24] R. R. Coifman and A. Sowa. Combining the calculus of variations and wavelets for image enhancement. Applied and Computational Harmonic Analysis, 9(1):1–18, 2000.

[25] P. L. Combettes. Solving monotone inclusions via compositions of nonexpansive averaged operators. Optimization, 53(5), Dec. 2004.

[26] P. L. Combettes and J.-C. Pesquet. A Douglas-Rachford splittting approach to nonsmooth convex variational signal recovery. IEEE Journal of Selected Topics in Signal Processing, 1(4):564–574, 2007.

[27] P. L. Combettes and V. R. Wajs. Signal recovery by proximal forward-backward splitting. SIAM Multiscale Model. Simul., 4(4):1168–1200, 2005.

[28] J. Darbon, M. Sigelle and F. Tupin. The use of levelable regularization functions for MRF restoration of SAR images while preserving reflectivity. Proc. of IS&T/SPIE 19th Annual Symposium Electronic Imaging, p. 6490, San Jose (USA), Feb. 2007.

[29] L. Denis, F. Tupin, J. Darbon and M. Sigelle. SAR Image Regularization With Fast Approximate Discrete Minimization. IEEE Transactions on Image Processing, 18(7):1588–1600, July 2009.

[30] D. L. Donoho and I. M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. Biometrika, 81(3):425–455, 1994.

[31] D. L. Donoho and I. M. Johnstone. Adapting to unknown smoothness via wavelet shrinkage. Journal of Acoustical Society America, 90, 1995.

[32] S. Durand and M. Nikolova. Denoising of frame coefficients using l_1 data-fidelity term and edge-preserving regularization. SIAM Journal on Multiscale Modeling and Simulation, 6(2):547–576, 2007.

[33] S. Durand and J. Froment. Reconstruction of wavelet coefficients using total variation minimization. SIAM Journal on Scientific Computing, 24(5):1754–1767, 2003.

[34] J. Eckstein and D. P. Bertsekas. On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators. Math. Programming: Series A and B, 55(3):293–318, July . 1992.

[35] J. Eckstein and B. F. Svaiter. A family of projective splitting methods for the sum of two maximal monotone operators. Math. Program., Ser. B, 111(1), Jan. . 2008.

[36] J. Froment and S. Durand. Artifact free signal denoising with wavelets. In Proceedings of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing, volume 6, 2001.

[37] S. Fukuda and H. Hirosawa. Suppression of speckle in synthetic aperture radar images using wavelet. Int. J. Remote Sens., 19(3):507–519, 1998.

[38] D. Gabay. Applications of the method of multipliers to variational inequalities. M. Fortin and R. Glowinski, editors, North-Holland, Amsterdam, 1983.

[39] J.-B. Hiriart-Urruty and C. Lemaréchal. Convex analysis and Minimization Algorithms, vol. I and II. Springer-Verlag, Berlin, 1996.

[40] Y.-M. Huang, M. K. Ng and Y.-W. Wen. A new total variation method for multiplicative noise removal. Technical report, Hong Kong Baptist University, http://www.math.hkbu.edu.hk/ICM/pdf/08-06.pdf, 2008.

[41] K. Krissian, C.-F. Westin, R. Kikinis and K. G. Vosburgh. Oriented speckle reducing anisotropic diffusion. IEEE Transactions on Image Processing, 16(5):1412–1424, May 2007.

[42] P.-L. Lions. Une méthode itérative de resolution d'une inéquation variationnelle. Israel Journal of Mathematics, 31(2):204–208, June . 1978.

[43] P.-L. Lions and B. Mercier. Splitting algorithms for the sum of two nonlinear operators. SIAM Journal on Numerical Analysis, 16(6):964–979, Dec. . 1979.

[44] F. Malgouyres. Mathematical analysis of a model which combines total variation and wavelet for image restoration. Journal of information processes, 2(1):1–10, 2002.

[45] F. Malgouyres. Minimizing the total variation under a general convex constraint for image restoration. IEEE Transactions on Image Processing, 11(12):1450–1456, Dec. 2002.

[46] J.-J. Moreau. Fonctions convexes duales et points proximaux dans un espace hilbertien. CRAS Sér. A Math., 255:2897–2899, 1962.

[47] P. Moulin and J. Liu. Analysis of multiresolution image denoising schemes using generalized gaussian and complexity priors. IEEE Transactions on Image Processing, 45(3):909–919, Apr. 1999.

[48] Y. Nesterov. Gradient methods for minimizing composite objective function. Technical report, Université catholique de Louvain, Center for Operations Research and Econometrics (CORE), CORE Discussion Papers 2007076, Sep. 2007.

[49] M. Nikolova. Local strong homogeneity of a regularized estimator. SIAM Journal on Applied Mathematics, 61(2):633–658, 2000.

[50] M. Nikolova. Minimizers of cost-functions involving nonsmooth data-fidelity terms. Application to the processing of outliers. SIAM Journal on Numerical Analysis, 40(3):965–994, 2002.

[51] M. Nikolova. A variational approach to remove outliers and impulse noise. Journal of Mathematical Imaging and Vision, 20(1), Jan. -Mar. 2004.

[52] M. Nikolova. Weakly constrained minimization. Application to the estimation of images and signals involving constant regions. Journal of Mathematical Imaging and Vision, 21(2):155–175, Sep. 2004.

[53] M. Nikolova. Analysis of the recovery of edges in images and signals by minimizing nonconvex regularized least-squares. SIAM Journal on Multiscale Modeling and Simulation, 4(3):960–991, 2005.

[54] G. B. Passty. Ergodic convergence to a zero of the sum of monotone operators in hilbert space. Journal of Mathematical Analysis and Applications, 72, 1979.

[55] A. Pizurica, A. M. Wink, E. Vansteenkiste, W. Philips and J.B.T.M. Roerdink. A review of wavelet denoising in MRI and ultrasound brain imaging. Current Medical Imaging Reviews, 2(2):247–260, 2006.

[56] B. Polyak. Introduction to Optimization. Optimization Software, Inc., Publications Division, New York, 1987.

[57] W.H. Press, S.A. Teukolsky, W.T. Vetterling and B.P. Flannery. Numerical recipes, the art of scientific computing. Cambridge Univ. Press, New York, 1992.

[58] R. T. Rockafellar. Monotone operators and the proximal point algorithm. SIAM Journal on Control and Optimization, 14(5):877–898, Aug. 1976.

[59] L. Rudin and S. Osher. Total variation based image restoration with free local constraints. Proc. IEEE ICIP, Austin, TX, vol. 1:31–35, Nov. . 1994.

[60] L. Rudin, P.-L. Lions and S. Osher. Multiplicative denoising and deblurring: Theory and algorithms., pages 103–119. Springer, Editors: S. Osher and N. Paragios, 2003.

[61] L. Rudin, S. Osher and C. Fatemi. Nonlinear total variation based noise removal algorithm. Physica, 60 D:259–268, 1992.

[62] J. Shi and S. Osher. A nonlinear inverse scale space method for a convex multiplicative noise model. SIAM J. Imaging Sciences, 1(3):294–321, 2008.

[63] E. P. Simoncelli. Bayesian denoising of visual images in the wavelet domain. Lecture Notes in Statistics, Vol. 41. Springer Verlag: Berlin, 1999.

[64] E. P. Simoncelli and E. H. Adelson. Noise removal via Bayesian wavelet coding. In Proceedings of the IEEE International Conference on Image Processing, pages 379–382, Lausanne, Switzerland, Sep. 1996.

[65] P. Tseng. Applications of a splitting algorithm to decomposition in convex programming and variational inequalities. SIAM Journal on Control and Optimization, 29(1):119–138, 1991.

[66] P. Tseng. A modified forward-backward splitting method for maximal monotone mappings. SIAM Journal on Control and Optimization, 38(1):431–446, 2000.

[67] M. Tur, C. Chin and J.W. Goodman. When is speckle noise multiplicative? Applied Optics, 21(7):11571159, April 1982.

[68] F. Ulaby and M. C. Dobson. Handbook of Radar Scattering Statistics for Terrain. Norwood, MA: Artech House, 1989.

[69] C. R. Vogel and M. E. Oman. Iterative method for total variation denoising. SIAM Journal on Scientific Computing, 17(1):227–238, 1996.

[70] G. Wang, J. Zhang and G.-W. Pan. Solution of inverse problems in image processing by wavelet expansion. IEEE Transactions on Image Processing, 4(5):579–593, May 1995.

[71] P. Weiss, L. Blanc-Féraud and G. Aubert. Efficient schemes for total variation minimization under constraints in image processing. SIAM Journal on Scientifigc Computing, (to appear) 2009.

[72] H. Xie, L. E. Pierce and F. T. Ulaby. SAR speckle reduction using wavelet denoising and markov random field modeling. IEEE Trans. Geosci. Remote Sensing, 40(10):2196–2212, Oct. 2002.

[73] Y. Yu and S. T. Acton. Speckle reducing anisotropic diffusion. IEEE Transactions on Image Processing, 11(11):1260–1270, Nov. 2002.

[74] M. Zhu, S.J. Stephen J. Wright and T.F. Chan. Duality-based algorithms for total-variation-regularized image restoration. Technical report, CAM 08-33, UCLA October, 2008.